

AD-A104 631

TECHNICAL  
LIBRARY

AD ~~A104631~~

MEMORANDUM REPORT ARBRL-MR-03126

NUMERICAL DIFFERENTIATION OF NOISY DATA

✓  
C. Masaitis  
G. Francis

August 1981



US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND  
BALLISTIC RESEARCH LABORATORY  
ABERDEEN PROVING GROUND, MARYLAND

Approved for public release; distribution unlimited.

Destroy this report when it is no longer needed.  
Do not return it to the originator.

Secondary distribution of this report by originating  
or sponsoring activity is prohibited.

Additional copies of this report may be obtained  
from the National Technical Information Service,  
U.S. Department of Commerce, Springfield, Virginia  
22161.

The findings in this report are not to be construed as  
an official Department of the Army position, unless  
so designated by other authorized documents.

*The use of trade names or manufacturers' names in this report  
does not constitute endorsement of any commercial product.*

**UNCLASSIFIED**

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER MEMORANDUM REPORT ARBRL-MR-03126	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) NUMERICAL DIFFERENTIATION OF NOISY DATA		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) C. Masaitis G. Francis		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS US Army Ballistic Research Laboratory ATTN: DRDAR-BLB Aberdeen Proving Ground, MD 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS RDT&E 1L161102AH43
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Armament Research & Development Command US Army Ballistic Research Laboratory ATTN: DRDAR-BL, APG, MD 21005		12. REPORT DATE AUGUST 1981
		13. NUMBER OF PAGES 59
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Approximation, Numerical Differentiation, Estimation of Derivatives, Smoothing, Curve Fitting, Differentiation of Noisy Data. <i>differentiation equations Run Meth &amp; Proc Data Reduction</i>		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) (hm) An algorithm has been developed based on the assumption that a given set of data is representable by a function that is a product of algebraic, trigonometric, and exponential polynomials. The degrees of the underlying polynomials, the optimal data spacing (optimal multiple of stepsize in tabular data), and the parameters of the approximating function are all selected by the algorithm to minimize an objective function that depends on the variance of the deviation between the data and the fitted function and on the degree of ill-conditioning of		

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(*When Data Entered*)

the problem. The algorithm permits choosing global or local fits. Derivatives of any order are obtained either by differentiating the fitted function or by a closed form relation between the model parameters and coefficients of linear combinations that express derivatives in terms of the observed values of the variable. Thus, the algorithm provides local differentiation formulas using parameters derived from global analysis of the data but in terms of the local values of the function. An appendix compares the accuracy of derivatives obtained for synthetic data with the accuracy of other methods.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(*When Data Entered*)

## TABLE OF CONTENTS

	Page
1. INTRODUCTION. . . . .	5
2. APPROXIMATION OF TABULAR FUNCTIONS AND THEIR DERIVATIVES . . . . .	9
2.1 Data Structure . . . . .	9
2.2 Smoothing and Differentiation. . . . .	16
3. DERIVATIVES AS LINEAR COMBINATIONS OF FUNCTIONAL VALUES . . . . .	18
4. DERIVATIVES WITH SIMPLE EIGENVALUES . . . . .	20
5. ERROR BOUNDS. . . . .	27
REFERENCES. . . . .	34
APPENDIX A. . . . .	35
APPENDIX B. . . . .	51
REFERENCE . . . . .	55
DISTRIBUTION LIST . . . . .	57

## 1. INTRODUCTION

Frequently data analysis and mathematical modeling require computation of derivatives of functions given at discrete points only.

For instance, temperature dependent heat capacity can be determined<sup>1</sup> by differentiating heat content. Dependence of the rate of plant photosynthesis on light intensity is computed<sup>2</sup> by measuring the amount of absorbed substances and by differentiating collected data numerically. Another example is the rate of elimination of a drug from blood, obtained by differentiating the measured concentration at successive time instances. These are three examples from rather diverse fields of inquiry. There are numerous investigations at BRL also requiring differentiation of tabular data. One example of such a problem is computation of target speed and acceleration from measured position coordinates.<sup>3</sup>

Another is computing a drag coefficient<sup>4</sup> from the differential equations of a flight trajectory by substituting velocity and acceleration values computed by differentiating position coordinates derived from radar data. Also the speed and acceleration of a projectile in the bore of a gun have been determined<sup>5</sup> by computing derivatives of discrete position data obtained by a microwave interferometer. Further examples of BRL research requiring differentiation of tabular data include determination of the slope of a shock front from discrete position points, computation of lead angle for antitank guns, and others.

In the cases mentioned above the derivative was obtained by BRL researchers either by drawing a smooth curve through the plotted points and then measuring its slope, or by a moving polynomial arc technique, or sometimes by use of a mechanical spline.

There are numerous open literature publications that discuss various numerical methods for computing derivatives of tabular functions.

<sup>1</sup>T.F. Dolgopolova and V.K. Ivanov, On Numerical Differentiation. *Zh. Vychisl Mat. Fiz.* 6, 3, 570-576, 1966.

<sup>2</sup>R.S. Anderssen and P. Bloomfield, "A Time Series Approach to Numerical Differentiation." *Technometrics* 16, 1 69-75, 1974.

<sup>3</sup>C. Masaitis, G. Francis and V. Woodward, "Survival vs. Horsepower per Ton Test Data Analysis," BRL MR 2518, 1975. (AD #C003211L)

<sup>4</sup>R.F. Lieske and A.M. MacKenzie, "Determination of Aerodynamic Drag from Radar Data," BRL MR 2210, 1972.

<sup>5</sup>F. Yagi, B. Jansen, L.B. Kennedy, W.C. Taylor, "Analysis of Interferometer Records of Projectile Motion in the Bore of Small Arms," BRL Report 939, 1955. (AD #74976)

Some of these simply differentiate polynomial interpolation formulas;<sup>6</sup> others use least squares fits of the data by trigonometric<sup>7</sup> or algebraic<sup>8</sup> polynomials. Still others use polynomial splines.<sup>9</sup> This last method imposes certain smoothness conditions on the fitted functions, and similar conditions are introduced by applying Tikhonov's<sup>10</sup> regularization procedure.<sup>1,11,12</sup>

Smoothness conditions of the spline functions and of the regularization procedure attempt to overcome the sensitivity of computed derivatives to small perturbation, i.e., the ill-posedness of the differentiation problem. Because of this sensitivity simple methods such as central difference or moving polynomial arc seldom produce satisfactory values of derivatives of tabular data containing measurement errors. Under these circumstances much more complicated methods are required, such as those of regularization, spline, or the method described in this report.

Since tabular data contain information about the function at discrete points only, and a derivative is the limit of a certain ratio, it cannot be obtained from numerical data without additional assumptions. The most common assumption is that tabular data are values of a certain function which can be identified on the basis of the data and subsequently differentiated to yield the values of the derivative. For instance, the approximation of a derivative by the divided difference assumes that tabular data can be represented by a straight line, at least locally. An extension of this assumption is obtained by fitting

<sup>6</sup>D.B. Hunter, "An Iterative Method of Numerical Differentiation," Comp. J. 3, 270-271, 1960.

<sup>7</sup>A. Talmi and G. Gilat, "Method for Smooth Approximation of Data," Journal of Comp. Physics 23, 93-123, 1977.

<sup>8</sup>H.C. Hershey, J.L. Zakin and R. Simha, "Numerical Differentiation of Equally Spaced and Not Equally Spaced Experimental Data," Ind. Eng. Chem. Fundam. 6, 413-421, 1967.

<sup>9</sup>P. Craven and G. Wahba, "Smoothing Noisy Data with Spline Functions," Numer. Math. 31, 377-403, 1979.

<sup>10</sup>A.N. Tikhonov, "Solution of Incorrectly Formulated Problems and the Regularization Method," Soviet Math. Dokl. 4, 1035-38, 1963.

<sup>11</sup>J. Cullum, "Numerical Differentiation and Regularization," Siam J. Numer. Anal. 8, 254-265, 1971.

<sup>12</sup>R.S. Anderssen and P. Bloomfield, "Numerical Differentiation Procedures for Non-Exact Data," Numer. Math. 22, 157-182, 1974.

a polynomial of prescribed degree and then differentiating it. For computational convenience, orthogonal polynomials may be selected<sup>8</sup> and data represented by a linear combination of these polynomials. The moving polynomial arc procedure assumes that the data can be locally represented by a polynomial of a specified degree. Thus, this procedure is, in a certain sense, a generalization of the divided difference approximation, which represents the data by a first degree polynomial fitted to two data points. Representation of data by spline functions, which are polynomials on the intervals between the nodes, also belongs to the same class, with a special data fitting criterion assuring that the  $L_2$  norm of a derivative of certain order is sufficiently small.

Besides polynomials and splines, frequently trigonometric polynomials and linear combinations of exponentials (exponential polynomials) are used to approximate tabular data, and then their derivatives are computed by differentiating the approximating polynomials. Also, approximation of tabular data by a linear combination of arbitrary orthogonal functions is frequently discussed in the literature.<sup>7</sup> However, in applications these functions are nothing more than Legendre or Chebychev polynomials, trigonometric functions, or just powers of the independent variable that are not even mutually orthogonal.

Thus almost all the approximations encountered in practice are least squares fits by algebraic, trigonometric, or exponential polynomials, either of a global type or local as with a moving polynomial arc, and either without additional constraints or with certain constraints as in the case of the spline procedure. Consequently, all these methods can be generalized by choosing an approximating function from the algebra, call it  $A$  for convenience, generated by algebraic, trigonometric, and exponential polynomials and by properly formulating an approximation criterion.

Usually the type of linear combination of selected functions is chosen in advance and only the weighting factors of this combination are adjusted to the data according to a selected criterion (mostly the least sum of residuals squared). Such a choice presupposes that the type of the function represented by the tabular data is known independently of the data, and the latter are used only to determine the weighting factors. A rather rare deviation from this procedure compares the reduction of the root mean squared error (RMSE) of the fits by polynomials of increasing degree and selects an approximating polynomial accordingly.<sup>7</sup> In this report a procedure is described for automatically selecting an element of the algebra  $A$  that minimizes the RMSE in an autoregressive model subject to constraints implied by three additional assumptions.

All the approximations mentioned above yield functional values  $y(t)$  as linear combinations of tabular data,  $x_j$ 's, with the weighting factors being linear combinations of the selected basis functions such

as polynomials, exponentials, etc., say  $\phi_1(t), \phi_2(t), \dots, \phi_k(t)$ :

$$y(t) = \sum_j x_j \sum_{i=1}^k c_{ji} \phi_i(t). \quad (1.1)$$

The coefficients  $c_{ji}$  may be independent of the data as in moving polynomial arc or regression procedures, or they may be rather complex functions of data points as in regularization and spline methods. Differentiation of (1.1) yields the derivative of the approximating function as a linear combination of data values:

$$y'(t) = \sum_j x_j \sum_i c_{ji} \phi'_i(t). \quad (1.2)$$

The weighting factors in (1.2) depend on  $t$ , i.e., on the position of the argument  $t$  relative to the data points,  $t_j$ 's, used to determine the coefficients  $c_{ji}$ . If the position of  $t$  is fixed, say, at the midpoint of the span of  $t_j$ 's, then the coefficients  $c_{ji}$  are independent of  $t$ . However, in either case the derivative is expressed as a linear combination of the data values, just as in the special case of its approximation by a divided difference.

The regularization procedure<sup>1,11,12</sup> obtains a numerical derivative by selecting an element in a Sobolev space by minimizing the sum of residuals squared plus a term proportional to the square of the norm of the approximating function. Thus, this is an extension of the spline approximation which considers only a particular norm of Sobolev space.<sup>9</sup> Furthermore, in practical applications the regularization procedure is restricted to readily computable elements of the Sobolev space and typically considers only trigonometric polynomials.<sup>11,12</sup>

In view of this it is natural to consider only representations of derivatives of tabular functions by linear combinations of the data values and to determine the weighting factors of such a combination directly from the data with a minimal set of assumptions. This is the approach of the present report.

Our basic assumption is that the given data represent an element of the algebra  $A$  as defined above. A criterion as to what constitutes the best representation of the data is our second assumption. It depends on the sum of the squares of certain residuals and on the degree of ill-conditioning of the problem. The definition of this criterion is contained in the next section. The criterion yields only the structure of an approximating element. The values of the element parameters can be subsequently determined by a linear least squares fit to a selected segment of the data. A derivative of any order is obtained either by differ-

entiating such a fitted approximation or by computing the weighting factors directly from the structure of the element.

Our third assumption is on the number of data values (and weighting factors) in the linear representation of a derivative. It is shown in this report that the derivative of every element of  $A$  is representable as a linear combination of its values and, conversely, a function is in  $A$  if its derivative can be represented as a linear combination of a finite number of its equally spaced values from any of the several data intervals.

Our fourth assumption is that the step size in the tabular data is sufficiently small so that the structure of the corresponding element of  $A$  is uniquely defined by the values of the data. This assumption implies that the selected element of  $A$  is sufficiently smooth, with the degree of smoothness determined by the step size and the accuracy of the data.

These four assumptions, spelled out precisely in the following sections, yield values of the derivatives that are sufficiently faithful to the empirical data and at the same time provide good approximations of derivatives when applied to synthetic data. Sections 2 and 3, together with Appendix A, present formal derivations of approximating formulas for both tabular data and related derivatives, first determining suitable structure based on eigenvalues and then finding parameters by least squares procedures. Section 4 deals with the frequently encountered case in which no eigenvalue is repeated. Here the formulas for the weighting coefficients can be simplified and rewritten in closed form in terms of real elements only, with a consequent reduction in round-off error as well as enhanced practical convenience. Section 5 presents formulas for approximating theoretical error bounds for derivatives computed by the procedures of Section 4. Appendix B illustrates the accuracy of the method and compares it to the accuracy of derivatives obtained by other commonly used methods.

## 2. APPROXIMATION OF TABULAR FUNCTIONS AND THEIR DERIVATIVES

As stated in the introduction we approximate tabular data by an element of the algebra  $A$  generated by algebraic, trigonometric, and exponential polynomials of a real variable  $t \in [0, T]$  with real coefficients. An approximating element of  $A$  is selected in two steps, by first determining its structure, most appropriate in a certain sense, and next by determining its parameters by a least squares fit.

### 2.1 Data Structure

The structure of an element of  $A$  approximating tabular data is obtained by observing that a function which satisfies a family of linear

difference equations dependent on the scaling of the independent variable is contained in  $A$  and, conversely, every element of  $A$  satisfies a linear difference equation of a fixed order with the coefficients dependent on the scaling of the independent variable.

By Demoivre's theorem trigonometric polynomials can be expressed as linear combinations of expressions  $(\cos \theta_j \pm i \sin \theta_j)^t = \lambda_j^t$  with  $\lambda_j = e^{\pm i \theta_j}$ , i.e., a trigonometric polynomial is a linear combination of exponentials with complex bases. Similarly, exponential polynomials

are of the form  $\sum c_j e^{\theta_j t} = \sum c_j \lambda_j^t$ . Consequently, every element of  $A$  is representable in the form

$$f(t) = \sum_{j=1}^m \sum_{p=0}^{n_j} c_{jp} \lambda_j^t t^p \quad (2.1)$$

with  $\lambda_j$ 's either complex or real. To every complex  $\lambda_j$  in (2.1) there corresponds its complex conjugate, say,  $\lambda_{j+1}$  with  $n_{j+1} = n_j$ . Let

$\Delta > 0$  and write (2.1) in the form

$$f(t) = \sum_{j=1}^m \sum_{p=0}^{n_j} c'_{jp} \left(\frac{t}{\Delta}\right)^p \left(\lambda_j^\Delta\right)^{\frac{t}{\Delta}}, \quad (2.2)$$

where  $c'_{jp} = c_{jp} \Delta^p$ . Define the polynomial  $P_\Delta(\lambda)$  of degree  $k$  where

$$k = \sum_{j=1}^m (n_j + 1) \quad (2.3)$$

as follows

$$P_\Delta(\lambda) = \prod_{j=1}^m (\lambda - \lambda_j^\Delta)^{n_j + 1}. \quad (2.4)$$

Let the operator  $B_\Delta$  be defined by  $B_\Delta f(t) = f(t-\Delta)$ . Then it follows from the properties of linear difference equations with constant coefficients that  $f(t)$  given by (2.2) satisfies the  $k$ -th order difference equation

$$P_\Delta(B_\Delta) f(t) = 0, \quad (2.5)$$

i.e., if  $f(t) \in A$  then for every  $\Delta > 0$   $f(t)$  satisfies the difference

equation of the form (2.5) defined by the  $\lambda_j$ , their multiplicities  $n_j+1$ , and  $\Delta$ . This can also be shown by substituting (2.1) in (2.5).

The converse is also true, i.e. if

- (a)  $y(t)$  satisfies (2.5) for every  $\Delta > 0$  and some  $\lambda_j$ 's and  $n_j+1$ ,
- (b)  $y(t)$  has bounded derivatives up to order  $k$  on the interval  $[0, T]$

then  $y(t) \in A$ . Without loss of generality we can assume that

- (c)  $y(t)$  satisfies no linear difference equation of order less than the  $k$  given by (2.3). The fact that such a  $y(t)$  is in  $A$  follows from the following proposition:

Proposition 1. If  $y(t)$  satisfies conditions (a), (b), and (c), then it is a solution of the differential equation

$$P(D)y(t) = 0 \quad (2.6)$$

where  $D = \frac{d}{dt}$  and

$$P(\xi) = \prod_{j=1}^m (\xi - \log \lambda_j)^{n_j+1} \quad (2.7)$$

Proof: By writing  $y^{(k)} = \lim_{\Delta \rightarrow 0} \frac{(1-B_\Delta)^k y}{\Delta^k}$  and by applying Lemma 1 in

Appendix A we have  $y^{(k)} = \frac{1}{k!} \frac{d^{(k)} N(\Delta)}{d\Delta^k} \Big|_{\Delta=0}$ . Hence by (A.10) and

$$\text{Lemma 3 of Appendix A} \quad y^{(k)}(t) = \sum_{j=1}^k (-1)^{j-1} \tilde{S}_{k,j} y^{(k-j)}(t).$$

Consequently, by the definition of  $\tilde{S}_{k,j}$  (see Lemma 3),  $y(t)$  satisfies (2.6).

In view of the relation between the algebra  $A$  and the equation (2.5) an element of  $A$  that approximates tabular data can be selected by constructing an appropriate autoregressive model by, say, a procedure developed for time series analysis<sup>13</sup>. However, this requires an assumption

<sup>13</sup> G.E. Box and G.M. Jenkins, Time Series Analysis: Forecasting and Control. Revised Edition. Holden-Day, 1976.

tion of stationarity or else instead of the original data, say  $x(n)$ ,  $n=0,1,\dots,N$ , one must approximate differences  $\Delta^d x(n)$  of order  $d$  that are assumed to be stationary. However, this assumption need not hold for tabular data that must be differentiated. For instance, if the underlying function is exponential the differences of any order are exponential and hence non-stationary. An example of this type of data is a concentration of injected drug in blood measured as a function of time after injection. In view of this, instead of attempting to determine the order of differences that may produce stationary series and at the same time considering possible periodicity ("seasonal" variation), we choose a direct method for estimating the coefficients of an autoregressive model. To this end we write  $x(n,p,q)$  for  $x(p+qn)$  where  $q$  is any positive integer and  $p=0,1,2,\dots,q-1, n=0,1,\dots,N_p$ , where  $N_p = \lfloor (N-p)/q \rfloor$  is the largest integer not exceeding  $(N-p)/q$ . Let  $y(t) \in A$  be an approximating function of the data, i.e.

$$x(n,p,q) = y(r\Delta) + \varepsilon_r, \quad (2.8)$$

where  $r=p+qn$  and  $\varepsilon_r$  is an observation error assumed to be weakly stationary white noise with zero mean and variance  $\sigma^2$ . The function  $y(t)$  satisfies (2.5) for a suitable polynomial  $P$ , say, of degree  $k$  since  $y(t) \in A$ . We write this equation as follows

$$y(r\Delta) = \sum_{j=1}^k a_j y[(r-jq)\Delta], \quad (2.9)$$

where the  $a_j$ 's remain to be determined. By substituting (2.8) in (2.9) we get

$$x(n,p,q) - \varepsilon_r = \sum_{j=1}^k a_j [x(n-j,p,q) - \varepsilon_{r-j}]. \quad (2.10)$$

By transposing the terms in (2.10) and squaring both sides we get:

$$\begin{aligned} & \left[ x(n,p,q) - \sum_{j=1}^k a_j x(n-j,p,q) \right]^2 \\ &= \varepsilon_r^2 + \sum_{j=1}^k a_j^2 \varepsilon_{r-j}^2 + P_r, \end{aligned} \quad (2.11)$$

where  $P_r$  is a linear combination of products  $\varepsilon_v \varepsilon_u$  with  $u \neq v$ . Since by assumption  $\varepsilon_r$  is white noise, we have  $E(P_r) = 0$ . Thus, by taking expected values of both sides of (2.11) we get:

$$\begin{aligned}
& E \left\{ \left[ x(n, p, q) - \sum_{j=1}^k a_j x(n-j, p, q) \right]^2 \right\} \\
& = \sigma^2 + \sigma^2 \sum_{j=1}^k a_j^2. \tag{2.12}
\end{aligned}$$

We replace the expected value of the left hand side by its estimate (average) and get:

$$\begin{aligned}
& \frac{1}{\tilde{N}} \sum_{p=0}^{q-1} \sum_{n=k+1}^{N_p} \left[ x(n, p, q) - \sum_{j=1}^k a_j x(n-j, p, q) \right]^2 \\
& = \sigma^2 + \sigma^2 \sum_{j=1}^k a_j^2, \tag{2.13}
\end{aligned}$$

where  $\tilde{N} = \sum_{p=0}^{q-1} (N_p - k)$ . Thus we get from (2.13):

$$\sigma^2 = \frac{\frac{1}{\tilde{N}} \sum_{p=0}^{q-1} \sum_{n=k+1}^{N_p} \left[ x(n, p, q) - \sum_{j=1}^k a_j x(n-j, p, q) \right]^2}{1 + \sum_{j=1}^k a_j^2}. \tag{2.14}$$

We choose the estimates of the  $a_j$ 's that minimize the variance  $\sigma^2$ . These estimates are obtained by the following iterative procedure. Let  $a$  denote the vector  $(a_j)$ . Assume that the value of  $a$ , say  $a^{(u)}$ , has been obtained on the  $u$ -th iteration. By substituting this value in (2.14) we obtain an estimate  $\sigma_u^2$  of the variance of  $\epsilon_r$ . Substitution of this value for  $\sigma^2$  in the second term of the right hand side of (2.13) yields:

$$\begin{aligned}
\sigma^2 & = \frac{1}{\tilde{N}} \sum_{p=0}^{q-1} \sum_{n=k+1}^{N_p} \left[ x(n, p, q) - \sum_{j=1}^k a_j x(n-j, p, q) \right]^2 \\
& - \sigma_u^2 \sum_{j=1}^k a_j^2. \tag{2.15}
\end{aligned}$$

The next estimate  $a^{(u+1)}$  of the vector  $a$  is obtained by minimizing the right hand side of (2.15). Partial derivatives of this expression with respect to the  $a_j$ 's when set to zero and multiplied by  $\tilde{N}/2$  yield

$$\begin{aligned} & \sum_{j=1}^k a_j \sum_{p=0}^{q-1} \sum_{n=k+1}^{N_p} x(n-j, p, q) x(n-i, p, q) - \tilde{N} \sigma_u^2 a_i \\ &= \sum_{p=0}^{q-1} \sum_{n=k+1}^{N_p} x(n, p, q) x(n-i, p, q), \end{aligned} \quad (2.16)$$

$i=1, 2, \dots, k$ . Thus, if  $M$  is the matrix of the normal equations of the overdetermined system

$$\sum_{j=1}^k a_j x(n-j, p, q) = x(n, p, q), \quad (2.17)$$

$n=k+1, \dots, N_p$ ,  $p=0, 1, \dots, q-1$ , and  $X$  is the right hand side vector of these normal equations, then the vector  $a^{(u+1)}$  that minimizes  $\sigma^2$  in (2.15) satisfies the equation

$$(M - \tilde{N} \sigma_u^2 I) a^{(u+1)} = X, \quad (2.18)$$

where  $I$  is the  $k \times k$  identity matrix. Successive iterates of the vector  $a$  are obtained by substituting  $\sigma_0^2 = 0$  in (2.18) and then by iterating (2.14) and (2.18). Thus, we need to obtain the normal equations of the overdetermined system (2.17) and then solve the linear equations (2.18).

In order to compute the  $a_j$ 's by this procedure we have to choose  $k$  and the integer  $q$  in (2.17). Obviously a larger number of model parameters, i.e. larger  $k$ , yields a model better matching the data. A smaller value of  $q$  provides a larger number of data points, i.e. describes the structure of the data more accurately. However, increasing  $k$  as well as reducing  $q$  makes the system (2.18) ill conditioned. Hence the value of  $k$  and the data spacing parameter  $q$  must be chosen to minimize  $\sigma^2$  in (2.14) and at the same time to prevent the matrix in (2.18) from becoming nearly singular. Thus, we have two conflicting criteria for selecting\* the optimal pair  $(k, q)$ . As usual, a measure

\* This is similar to the solution of the numerical differentiation problem by regularization<sup>11</sup> where increasing the regularization parameter reduces ill-conditioning of the problem and decreasing the parameter yields a better fit of the data.

of optimality must be chosen heuristically. Our choice is an index

$$J(k, q) = \sigma^2(k, q) / [D(k, q)]^{\sqrt{k}}, \quad (2.19)$$

where  $D(k, q)$  is the absolute value of the determinant of the last iteration of (2.18) corresponding to the choice of  $k$  and  $q$ . Thus, we compute the  $a_j$ 's and  $J(k, q)$  for  $k=1, 2, \dots, k_0$  and  $q=1, 2, \dots, q_0$  and select the pair  $(k, q)$  and the corresponding  $a_j$ 's that minimize  $J(k, q)$ . We impose an additional constraint on  $(k, q)$  in order to prevent a choice of a model for which the data are inadequate, i.e. a model that contains terms of higher frequency than can be determined by the frequency of the data points. Thus, if  $w_m$  is the maximum frequency of the selected model then we must at least have

$$q \Delta < \frac{2\pi}{w_m} \quad (2.20)$$

Suppose further that for some  $q$  the coefficients,  $a_j$ 's, in (2.17) yield a real negative eigenvalue, say  $\lambda_j < 0$ . Then the term  $c_{j0} \lambda_j^n$  in (2.1) is equal to  $c_{j0} |\lambda_j|^n \cos n\pi$  for every  $n$ . The frequency of this term is  $\pi$  radians per  $q\Delta$  sec or  $\pi/q\Delta$  radians per sec, i.e. we have  $w_m \geq \pi/q\Delta$ , contrary to requirement (2.20).

If for some  $k$  and  $q$  the equation (2.17) has an eigenvalue with a negative real part, say  $\lambda_j = -a+ib$  ( $a>0$ ), then the corresponding term  $c_{j0} \lambda_j^n$  in (2.1) is expressible as  $c_{j0} \exp(\rho n + i\omega n)$  where  $\cos \omega = -a/\sqrt{a^2+b^2}$ , i.e.  $\omega > \frac{\pi}{2}$  if expressed in radians per unit time equal to  $q\Delta$  sec. Therefore this choice of  $q$  yields a spacing  $q\Delta$  with less than four data points per period of the corresponding term in (2.1). Although theoretically two points per period may be adequate to determine the real and imaginary parts of the corresponding eigenvalue, even three points per period are inadequate when the data contain measuring errors. Furthermore, a negative real part only implies that the corresponding frequency is greater than  $\pi/2$  per unit time. It may also be greater than  $\pi$  and less than  $3\pi/2$ , in which case the spacing  $q\Delta$  provides less than two points per period. This is the reason why the pairs  $(k, q)$  leading to complex roots with negative real part are rejected.

In summary, the models (2.17) are determined for  $q=1, 2, \dots, q_0$ ,  $k=1, 2, \dots, k_0$  and among those with eigenvalues having non-negative real components that one which yields minimum  $J(k, q)$  in (2.18) is selected.

When the data is very noisy this selection of  $q$  may lead to a rather large step size  $q\Delta$  and, thus, may eliminate high frequency terms present in the data even if the original spacing  $\Delta$  is adequate to represent this high frequency. This may happen when the amplitudes of high frequency terms are too small relative to the measuring error  $\varepsilon_r$  to be determined by the data taken at any spacing  $\Delta$ . The procedure described above is intended to determine only the terms of (2.1) for which both the spacing and also the accuracy of the data are adequate, and this works satisfactorily in practice.

## 2.2 Smoothing and Differentiation

The procedure just described determines the coefficients  $a_j$  in equation (2.9), i.e. determines the structure of an element of  $A$  that minimizes  $J(k, q)$ . The corresponding element of  $A$  is then given by (2.1),

i.e. by  $y(t_0 + nq\Delta) = \sum_{j=1}^m \sum_{i=0}^{n_j} \tilde{c}_{ji} n^i \mu_j^n$ , where the  $\mu_j$ 's are the roots of

$$\mu^k = \sum_{j=1}^k a_j \mu^{k-j} \quad (2.21)$$

of multiplicity  $n_j+1$ . By reverting back from  $q\Delta$  to the original spacing  $\Delta$  of the data and by writing  $\lambda_j$  for  $\mu_j^{1/q}$  we can write (2.21) as follows:

$$\sum_{j=1}^m \sum_{i=0}^{n_j} c_{ji} u^i \lambda_j^u = x_u, \quad (2.22)$$

where  $c_{ji} = \tilde{c}_{ji}/q^i$ ,  $u = \frac{t}{\Delta}$ , and  $y(t_0 + t)$  is replaced by the data value  $x_u = x(t_0/\Delta + t/\Delta)$ .

The coefficients in (2.22) are selected to minimize the RMSE of the resulting approximation. This may be either a global or a local approximation over the span, say, from  $u = -K$  to  $u = K$ . In the latter case the coefficients  $c_{ji}$  are obtained by the least squares method from (2.22) with  $u = -K, -K+1, \dots, K$ . The corresponding normal equations are

$$\sum_{u=-K}^K \sum_{j=1}^m \sum_{i=0}^{n_j} c_{ji} u^{i+\alpha} \lambda_j^u \lambda_\beta^u = \sum_{u=-K}^K x_u u^\alpha \lambda_\beta^u$$

or in matrix notation

$$S C = Z. \quad (2.23)$$

Here  $S$  is a  $k \times k$  matrix ( $k = \sum_{j=1}^m (n_j + 1)$ ) with elements

$$S_{zv} = \sum_{u=-K}^K u^{i+\alpha} \lambda_j^u \lambda_\beta^u, \quad i=0,1,\dots,n_j; \quad j=1,2,\dots,m; \quad \alpha=0,1,\dots,n_\beta;$$

$\beta=1,2,\dots,m$ .  $C$  and  $Z$  are column vectors with the components  $c_{ji}$  and  $\sum_{u=-K}^K x_u u^\alpha \lambda_\beta^u$ , respectively. From (2.23) we get

$$C = S^{-1}Z. \quad (2.24)$$

Let  $\Lambda$  be a column vector with components  $v^i \lambda_j^v$ . Then substitution of  $C$  from (2.24) in (2.22) yields a smoothed value of  $x_v$

$$\hat{x}_v = \Lambda^T S^{-1} Z. \quad (2.25)$$

Let  $\gamma_u$  be the column vector with components  $u^\alpha \lambda_\beta^u$ . Then replacing  $Z$  in (2.25) we get:

$$\hat{x}_v = \sum_{u=-K}^K x_u \Lambda^T S^{-1} \gamma_u \quad (2.26)$$

where  $\Lambda^T S^{-1} \gamma_u$  depends only on the  $\lambda_j$ 's,  $u$  and  $v$  but not on  $x_u$ , i.e. (2.26) provides a smoothed value of the data point  $x_v$  as a linear combination of the data values  $x_u$ . If (2.26) is used to compute the smoothed value at the midpoint of the span of the data from  $-K$  to  $+K$ , (i.e.  $\hat{x}_0$ , then the weighting factors remain constant as the span, together with its midpoint, is translated along the axis of the independent variable  $t$ .

The vector  $C$  given by (2.24) can be used to compute derivatives of tabular data. By differentiating (2.1), rescaling the independent variable  $t$  as in (2.22), substituting  $C$  from (2.24), and interchanging matrix multiplication and summation we obtain the following expression for the derivative:

$$\hat{x}'_v = \frac{1}{\Delta} \sum_{u=-K}^K x_u \Lambda_1^T S^{-1} \gamma_u \quad (2.27)$$

where  $\Lambda_1 = \frac{d\Lambda}{du}$  is a column vector with components  $(u^i \lambda_j^u \log \lambda_j + i u^{i-1} \lambda_j^u)$ ,  $i=0,1,\dots,n_j$ ,  $j=1,2,\dots,m$ .

Similarly, higher derivatives are obtained by the formula

$$\hat{x}_v^{(p)} = \frac{1}{\Delta^p} \sum_{u=-K}^K x_u \Lambda_p^T S^{-1} \gamma_u \quad (2.28)$$

where  $\Lambda_p = \frac{d^p \Lambda}{du^p}$ .

Weighting factors in a formula like (2.28) for derivatives of various orders can be obtained directly from the structure of the data defined by the eigenvalues of a corresponding autoregressive model without computing the vector  $C$  in (2.24). Such a procedure is especially convenient when all the eigenvalues are simple, and this case is described in Section 4.

### 3. DERIVATIVES AS LINEAR COMBINATIONS OF FUNCTIONAL VALUES

It was mentioned in the introduction that all the common methods for numerical differentiation express derivatives as linear combinations of discrete functional values in the form

$$y'(t) = \sum_{i=1}^k a_i y[t + (p-i)\Delta], \quad (3.1)$$

where  $p$  is some integer and  $\Delta$  is the spacing between data points used. Various methods apply different criteria by which the coefficients  $a_i$  in (3.1) are selected. A generalization of all these methods could be obtained by identifying a set, say  $\tilde{A}$ , of functions that satisfy the relation such as (3.1) and by defining a set of criteria for selecting the coefficients. This we do below and in Section 4.

Obviously, any function that has a derivative at at least one point satisfies (3.1) for at least one value of  $\tilde{t}$ . Therefore we restrict the set  $\tilde{A}$  by insisting that every element of  $\tilde{A}$  satisfies (3.1) for all  $t$ 's in an interval such as  $[0, T]$ . If we allow the coefficients  $a_i$  to vary with  $t$  in a different fashion for every function in  $\tilde{A}$ , then every differentiable and non-vanishing function in  $[0, T]$  is in  $\tilde{A}$ , since we can write  $f'(t) = [f'(t)/f(t)] f(t)$  which is a special case of (3.1). Therefore we define the set  $\tilde{A}$  as a set of functions  $y(t)$  each of which satisfies (3.1) with  $a_i$ 's dependent only on  $y(t)$ ,  $p$ , and  $\Delta$  but not on  $t$ .

We choose in (3.1)  $p=k+s+1$  and replace the summation index  $i$  by  $-j+k+s+1$ . This yields

$$y'(t) = \sum_{j=s+1}^{k+s} a(j,s) y(t + j\Delta), \quad (3.2)$$

where  $a(j,s) = a_{-j+k+s+1}$ . Thus, the set  $\tilde{A}$  is defined as a set of functions that satisfy (3.2) for every  $t \in [0, T]$  and every integer  $s$  such that  $-t - \Delta \leq s\Delta \leq T - k\Delta$ .

The following proposition is implied by Lemmas 4 and 5 in Appendix A:

Proposition 2.  $\tilde{A} = A$ , where  $A$  is the algebra generated by algebraic, trigonometric, and exponential polynomials over the interval  $[0, T]$ .

Proof: First we show that  $\tilde{A} \subset A$ . To this end we choose  $s=-1$  in (3.2) and write

$$y'(t) = \sum_{j=1}^k b_j^{(1)} y[t + (j-1)\Delta], \quad (3.3)$$

where  $b_j^{(1)} = a(j-1, -1)$ . We observe that

$$y^{(q)}(t) = \sum_{j=1}^k b_j^{(q)} y[t + (j-1)\Delta], \quad (3.4)$$

for  $q \leq k$  and some constants  $b_j^{(q)}$ . This relation is proved by induction on  $q$ . Indeed, (3.4) holds for  $q=1$  because of (3.3). Suppose (3.4) holds for  $q < k$ . Differentiation of (3.4) with respect to  $t$  yields:

$$y^{(q+1)}(t) = \sum_{j=1}^k b_j^{(q)} y'[t + (j-1)\Delta]. \quad (3.5)$$

Substitute  $t + h\Delta$  for  $t$  in (3.2), let  $s = -h-1$  and change the summation index  $j$  to  $u-h-1$ . This yields

$$y'(t+h\Delta) = \sum_{u=1}^k a(u-h-1, -h-1) y[t + (u-1)\Delta]. \quad (3.6)$$

It follows from (3.5) and (3.6) that  $y^{(p+1)}(t) = \sum_{j=1}^k b_j^{(p+1)} y[t + (j-1)\Delta]$ , where  $b_j^{(p+1)} = \sum_{u=1}^k b_u^{(p)} a(j-u, -u)$ , i.e.

(3.4) holds for  $q+1$  and consequently it holds for  $q=1, 2, \dots, k$ . Let  $B$  be the  $k \times k$  matrix with  $q^{\text{th}}$  row  $(b_1^{(q)}, b_2^{(q)}, \dots, b_k^{(q)})$ . If  $B$  is non-singular then we have from (3.4):

$$y(t) = \sum_{j=1}^k c_j y^{(j)}(t), \quad (3.7)$$

where the  $c_j$ 's are the corresponding elements of  $B^{-1}$ . If the rank of  $B$  is less than  $k$ , then there exist constants  $d_1, d_2, \dots, d_k$ , not all zero, such that

$$\sum_{q=1}^k d_q b_j^{(q)} = 0, \quad (3.8)$$

$j=1, 2, \dots, k$ . Hence it follows from (3.4) that

$$\sum_{q=1}^k d_q y^{(q)} = 0. \quad (3.9)$$

Consequently  $y(t) \in \tilde{A}$  satisfies either (3.7) or (3.9) and hence  $y(t) \in A$ , i.e.  $\tilde{A} \subset A$ .

The converse inclusion  $A \subset \tilde{A}$  follows from Lemmas 4 and 5 in Appendix A. Indeed, let  $y(t) \in A$ , i.e. let  $y(t)$  be of the form (A.20) as in Lemma 4. The lambdas in (A.20) define the matrix  $L$  of Lemma 4. By Lemma 5  $\det L \neq 0$ . Hence there exists a unique solution vector  $b$  of equation (A.21) in Lemma 4. Consequently,  $y(t)$  satisfies (A.22) and hence  $y(t) \in \tilde{A}$ , i.e.  $A \subset \tilde{A}$ . This completes the proof of the proposition.

Thus, if we either assume that tabular data can be approximated by an element of  $A$ , or that the derivative of the corresponding function can be expressed as a linear combination of certain data values, then in view of propositions 1 and 2 an approximating element can be obtained by constructing an appropriate autoregressive model such as (2.8) in the preceding section. In other words, the method described in the preceding section yields the exact derivative for a function in the set  $A = \tilde{A}$ , or, equivalently, in the set of functions that satisfy a family of difference equations of the type (2.4).

#### 4. DERIVATIVES WITH SIMPLE EIGENVALUES

Equations (A.21) and (A.22) in Appendix A yield derivatives of tabular functions representable by autoregressive models, including those with simple eigenvalues. The needed weights  $b_{j-s}$  in (A.22) are

obtained by solving the linear system of equations (A.21). This requires complex number arithmetic since the coefficients in the equation (A.21) are imaginary for imaginary eigenvalues  $\lambda_j$ .

A closed form expression in terms of components of complex eigenvalues can be obtained for the case of simple characteristic roots. An element of  $A$  corresponding to this case is

$$y(t) = \sum_{j=1}^k c_j \lambda_j^t, \quad (4.1)$$

i.e. it belongs to the subalgebra of  $A$  generated by trigonometric and exponential polynomials. The case of simple eigenvalues is of practical interest for the following reason.

Equality of two eigenvalues defines a functional relation between the  $a_j$ 's, the coefficients of the corresponding autoregressive model, i.e. it defines a hypersurface in the  $k$ -dimensional Euclidean space  $X^{(k)}$  of  $a_1, a_2, \dots, a_k$ . Thus, the set of vectors  $a$  in  $X^{(k)}$  that define multiple roots has Lebesgue measure zero. Therefore if we assume that experimental data yields points of  $a \in X^{(k)}$ , all equally likely in a finite region, i.e. with probability distribution proportional to the Lebesgue measure, then the probability of obtaining equal or nearly equal eigenvalues is very small, although synthetic data derived from algebraic polynomials does yield multiple roots.

In order to obtain a closed form solution of equation (A.21) we need the following lemmas.

Lemma 6. Let  $V_{n+1}$  be the Vandermonian of  $x_0, x_1, x_2, \dots, x_n$  and  $V_n^{(j)}$  be its minor corresponding to the element  $x_0^j$ . Then

$$V_n^{(j)} = S_{n-j} V_n^{(n)}, \quad (4.2)$$

where  $S_{n-j}$  is the symmetric function of  $x_1, x_2, \dots, x_n$  of order  $n-j$  and  $V_n^{(n)}$  is the Vandermonian of  $x_1, x_2, \dots, x_n$ .

Proof. By expanding  $V_{n+1}$  with respect to the first row (the row consisting of powers of  $x_0$ ) we get

$$V_{n+1} = \sum_{j=0}^n (-1)^j x_0^j V_n^{(j)}.$$

Divide both sides by  $(-1)^n v_n^{(n)}$ :

$$(-1)^n \frac{v_{n+1}}{v_n^{(n)}} = \sum_{j=0}^n (-1)^{j+n} x_0^j \frac{v_n^{(j)}}{v_n^{(n)}}. \quad (4.3)$$

Since  $v_n^{(n)}$  is itself a Vandermonian we have

$$v_n^{(n)} = \prod_{j=2}^n \prod_{i=1}^{j-1} (x_j - x_i). \text{ Also, } v_{n+1} = \prod_{j=1}^n \prod_{i=0}^{j-1} (x_j - x_i).$$

Therefore we get from (4.3)

$$\sum_{j=1}^n (x_0 - x_j) = \sum_{j=0}^n (-1)^{j+n} x_0^j \frac{v_n^{(j)}}{v_n^{(n)}}. \quad (4.4)$$

The right hand side of this identity is thus a polynomial in  $x_0$  of degree  $n$  with the roots  $x_1, x_2, \dots, x_n$ , and the coefficient of  $x_0^n$  is equal to one. Therefore the coefficient of  $x_0^j$  is  $(-1)^{n-j} s_{n-j}$ . Thus, we have from (4.4)  $s_{n-j} = v_n^{(j)}/v_n^{(n)}$ , which is (4.2).

Lemma 7. Let  $s_i$  be the symmetric function of  $\lambda_1, \lambda_2, \dots, \lambda_k$  of order  $i$  and  $s_i^{(p)}$  be the symmetric function of  $\lambda_1, \lambda_2, \dots, \lambda_{p-1}, \lambda_{p+1}, \dots, \lambda_k$  of order  $i$ . Then

$$s_i^{(p)} = \sum_{j=0}^i (-1)^j s_{i-j} \lambda_p^j. \quad (4.5)$$

Proof by induction on  $i$ . For  $i=0$  (4.5) is an identity. Suppose (4.5) holds for  $i < k$ . The function  $s_{i+1}$  can be expressed as the sum of all the products of  $i+1$  factors taken from  $\lambda_1, \lambda_2, \dots, \lambda_{p-1}, \lambda_{p+1}, \dots, \lambda_k$  plus the sum of all the products of  $i$  factors taken from the same set with each product multiplied by  $\lambda_p$ , i.e.  $s_{i+1} = s_{i+1}^{(p)} + s_i^{(p)} \lambda_p$  or  $s_{i+1}^{(p)} = s_{i+1} - s_i^{(p)} \lambda_p$ . By substituting  $s_i^{(p)}$  from (4.5) we get  $s_{i+1}^{(p)} = s_{i+1} - \sum_{j=0}^i (-1)^j s_{i-j} \lambda_p^{j+1}$ . Let  $j=u-1$ . Then

$s_{i+1}(p) = s_{i+1} - \sum_{u=1}^{i+1} (-1)^{u-1} s_{i-u+1} \lambda_p^u = \sum_{u=0}^{i+1} (-1)^u s_{i+1-u} \lambda_p^u$ . This is the relation (4.5) with  $i$  advanced to  $i+1$  and the proof is complete.

When all eigenvalues are simple equation (A.21) is of the form

$$\begin{pmatrix} \lambda_1^{s+1} & \lambda_1^{s+2} & \dots & \lambda_1^{s+k} \\ \lambda_2^{s+1} & \lambda_2^{s+2} & \dots & \lambda_2^{s+k} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_k^{s+1} & \lambda_k^{s+2} & \dots & \lambda_k^{s+k} \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{pmatrix} = \begin{pmatrix} \log r_{\lambda_1} \\ \log r_{\lambda_2} \\ \vdots \\ \log r_{\lambda_k} \end{pmatrix} \cdot \frac{1}{\Delta^r} \quad (4.6)$$

where  $\Delta$  is the spacing used in determining the  $\lambda$ 's (i.e.,  $q$  times the original data spacing). The solution  $b$  of (4.6) is obtained by Cramer's rule. This allows us to cancel common factors in the numerator and denominator that are differences of eigenvalues and, hence, to reduce the round-off error.

The determinant of (4.6) can be written as follows:

$$\det L = s_k^{s+1} v_k = \prod_{j=1}^k \lambda_j^{s+1} \prod_{j>i} (\lambda_j - \lambda_i). \quad (4.7)$$

Let

$$Q(\lambda_p) = \prod_{\substack{i=1 \\ i \neq p}}^k (\lambda_p - \lambda_i) \quad (4.8)$$

and

$$P(\lambda) = \sum_{j=0}^k a_j \lambda^{k-j} \quad (4.9)$$

( $a_0 = -1$ ) be the characteristic polynomial of (3.1). Then

$$\frac{\partial p}{\partial \lambda} \Big|_{\lambda=\lambda_p} = - \prod_{\substack{j=1 \\ j \neq p}}^k (\lambda_p - \lambda_j). \quad \text{Therefore}$$

$$Q(\lambda_p) = - \sum_{j=0}^{k-1} (k-j) a_j \lambda_p^{k-j-1} \quad (4.10)$$

or

$$Q(\lambda_p) = - \sum_{q=1}^k q a_{k-q} \lambda_p^{q-1}. \quad (4.11)$$

By writing  $\prod_{j>i} (\lambda_j - \lambda_i) = \prod_{\substack{i=1 \\ i \neq p}}^k (\lambda_p - \lambda_i) \prod_{\substack{j>i \\ i, j \neq p}} (\lambda_j - \lambda_i) (-1)^{k-p}$  and by

observing that the last product is a Vandermonian of  $\lambda_1, \lambda_2, \dots, \lambda_{p-1}, \lambda_{p+1}, \dots, \lambda_k$ , say  $\tilde{V}_{k-1}^{(p)}$ , we can write (4.7) as follows:

$$\det L = (-1)^{k-p} s_k^{s+1} Q(\lambda_p) \tilde{V}_{k-1}^{(p)}. \quad (4.12)$$

Continuing the application of Cramer's rule, let  $B_{j-s}$  be the matrix obtained from  $L$  by replacing its  $(j-s)$ -th column by the right hand side of (4.6). By expanding  $\det B_{j-s}$  with respect to this column we get:

$$\det B_{j-s} = \frac{(-1)^{j-s}}{\Delta} \sum_{p=1}^k (-1)^p \log r \lambda_p s_k^{s+1} \lambda^{-s-1} \tilde{V}_{k-1}^{(j-s-1)}(p),$$

where

$$\tilde{V}_{k-1}^{(j-s-1)}(p) = \begin{vmatrix} 1 & \lambda_1 & \dots & \lambda_1^{j-s-2} & \lambda_1^{j-s} \dots & \lambda_1^{k-1} \\ \hline \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & \lambda_{p-1} & \dots & \lambda_{p-1}^{j-s-2} & \lambda_{p-1}^{j-s} \dots & \lambda_{p-1}^{k-1} \\ 1 & \lambda_{p+1} & \dots & \lambda_{p+1}^{j-s-2} & \lambda_{p+1}^{j-s} \dots & \lambda_{p+1}^{k-1} \\ \hline \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & \lambda_k & \dots & \lambda_k^{j-s-2} & \lambda_k^{j-s} \dots & \lambda_k^{k-1} \end{vmatrix} \quad (4.14)$$

i.e.  $\tilde{V}_{k-1}^{(j-s-1)}(p)$  is a corresponding minor of  $\det B_{j-s}$  divided by  $s_k^{s+1} \lambda_p^{-s-1}$ .

Determinant (4.14) is a minor of a Vandermonian in  $\lambda_0, \lambda_1, \dots, \lambda_{p-1}, \lambda_{p+1}, \dots, \lambda_k$  corresponding to the  $(j-s-1)$ -th element of the first row. Thus, by Lemma 6 above we have:

$$v_{k-1}^{(j-s-1)}(p) = s_{k-j+s}^{(p)} \tilde{v}_{k-1}^{(p)}, \quad (4.15)$$

where  $s_{k-j+s}^{(p)}$  is the symmetric function of order  $k-j+s$  in  $\lambda_1, \lambda_2, \dots, \lambda_{p-1}, \lambda_{p+1}, \dots, \lambda_k$  and  $\tilde{v}_{k-1}^{(p)}$  is the Vandermonian of the same arguments.

By substituting (4.5) in (4.15) we get

$$v_{k-1}^{(j-s-1)}(p) = \tilde{v}_{k-1}^{(p)} \sum_{v=0}^{k-j+s} (-1)^v s_{k-j+s-v} \lambda_p^v. \quad (4.16)$$

Since  $s_i$  is a symmetric function of the roots of (4.9) we have

$$s_{k-j+s-v} = (-1)^{k-j+s-v-1} a_{k-j+s-v}. \quad (4.17)$$

By substituting (4.15) - (4.17) in (4.13) we get:

$$\begin{aligned} \det B_{j-s} = \frac{1}{\Delta^r} \sum_{p=1}^k (-1)^p \log^r \lambda_p s_k^{s+1} \lambda_p^{-s-1} v_{k-1}^{(p)} \\ \cdot \sum_{v=0}^{k-j+s} (-1)^{k-1} a_{k-j+s-v} \lambda_p^v. \end{aligned} \quad (4.18)$$

In view of (4.11), (4.12), and (4.13) Cramer's rule yields:

$$b_j = \frac{1}{\Delta^r} \sum_{p=1}^k \frac{\lambda_p^{-s} \log^r \lambda_p \sum_{v=0}^{k-j} a_{k-j-v} \lambda_p^v}{\sum_{v=1}^k v \lambda_p^v a_{k-v}}. \quad (4.19)$$

Formula (4.19) provides a closed form expression for the coefficients in (3.2) in terms of the coefficients  $a_j$  of the autoregressive model (2.9) and in terms of the eigenvalues of the corresponding difference equation. Since some of the eigenvalues may be imaginary, computation of the  $b_j$ 's by (4.19) still may require complex number arithmetic. This can be avoided by combining complex conjugate terms in (4.19).

Let  $(\lambda_1, \lambda_1^*), (\lambda_2, \lambda_2^*), \dots, (\lambda_u, \lambda_u^*)$  be the pairs of complex conjugate roots and  $\lambda_{2u+1}, \dots, \lambda_k$  be real roots of the characteristic polynomial.

Write (4.19) as follows:

$$b_j = \frac{1}{\Delta r} \sum_{p=1}^u \left[ \frac{\lambda_p^{-s} \log^r \lambda_p \sum_{v=0}^{k-j} a_{k-j-v} \lambda_p^v}{\sum_{v=1}^k v a_{k-v} \lambda_p^v} + T^* \right]$$

$$+ \frac{1}{\Delta r} \sum_{p=2u+1}^k \frac{\lambda_p^{-s} \log^r \lambda_p \sum_{v=0}^{k-j} a_{k-j-v} \lambda_p^v}{\sum_{v=1}^k v a_{k-v} \lambda_p^v},$$

where  $T^*$  is obtained from the first term in the brackets by replacing  $\lambda_p$  by  $\lambda_p^*$ . By adding the terms in the brackets we get

$$\frac{\left( \lambda_p^{-s} \log^r \lambda_p \sum_{v=0}^{k-j} a_{k-j-v} \lambda_p^v \right) \sum_{v=1}^k d_v \lambda_p^{*v} + \left( \lambda_p^{*-s} \log^r \lambda_p^* \sum_{v=0}^{k-j} a_{k-j-v} \lambda_p^{*v} \right) \sum_{v=1}^k d_v \lambda_p^v}{\left( \sum_{v=1}^k d_v \lambda_p^v \right) \left( \sum_{v=1}^k d_v \lambda_p^{*v} \right)}, \quad (4.20)$$

where  $d_v = v a_{k-v}$ .

The denominator  $D_p$  of the last expression can be transformed as follows:

$$D_p = \sum_{v=1}^k d_v^2 (\lambda_p \lambda_p^*)^v + \sum_{v \neq n} (d_v d_n) (\lambda_p^v \lambda_p^{*n} + \lambda_p^{*n} \lambda_p^v).$$

By writing  $\lambda_p = A_p + iB_p$ ,  $\alpha_p = \log |\lambda_p|$ , and  $\beta_p = \arctan B_p/A_p$  we get

$$D_p = \sum_{v=1}^k \sum_{n=1}^k e^{(v+n)\alpha_p} \cos[(v-n)\beta_p] d_v d_n.$$

Similarly, the numerator  $N_p$  of (4.20) can be transformed into

$$N_p = 2 \sum_{v=0}^{k-j} \sum_{n=1}^k a_{k-j-v} d_n \exp \left[ (v+n-s)\alpha_p + r \log \sqrt{\alpha_p^2 + \beta_p^2} \right].$$

$$+ \cos \left[ (v-n-s)\beta_p + r \arcsin \left( \beta_p / \sqrt{\alpha_p^2 + \beta_p^2} \right) \right].$$

Neither  $N_p$  nor  $D_p$  involves any imaginary elements.

Consequently, we get:

$$b_j = \frac{1}{\Delta^r} \sum_{p=1}^u \frac{N_p}{D_p} + \frac{1}{\Delta^r} \sum_{p=2u+1}^k \frac{\lambda_p^{-s} \log^r \lambda_p \sum_{v=0}^{k-j} a_{k-j-v} \lambda_p^v}{\sum_{v=1}^k v a_{k-v} \lambda_p^v}, \quad (4.21)$$

a closed form solution for the coefficients in (3.2).

## 5. ERROR BOUNDS

In this section we develop expressions which yield approximate bounds on the errors in derivatives calculated by the method of Section 4.

Let  $x_t = x(t)$ ,  $t=0, \Delta, 2\Delta, \dots, nq\Delta$  be tabular data of a function  $y(t) \in A$  with the measuring error  $\varepsilon_t$ , i.e.

$$x(t) = y(t) + \varepsilon_t. \quad (5.1)$$

We assume that  $\varepsilon_t$  is a white noise with zero mean and variance  $\sigma^2$ . As shown on the preceding pages the  $r$ -th derivative of  $y(t)$  is expressible in the form

$$y^{(r)}(t) = \sum_{j=s+1}^{k+s} b_{j-s}^{(r)} y(t+jq\Delta), \quad (5.2)$$

and its approximation  $x^{(r)}(t)$  is

$$x^{(r)}(t) = \sum_{j=s+1}^{k+s} \hat{b}_{j-s}^{(r)} x(t+jq\Delta), \quad (5.3)$$

where the  $\hat{b}_{j-s}$ 's are expressed in terms of the coefficients in an auto-regressive model

$$x(t) = \sum_{j=1}^k \hat{a}_j x(t-jq\Delta) \quad (5.4)$$

corresponding to the difference equation

$$y(t) = \sum_{j=1}^k a_j y(t-jq\Delta) \quad (5.5)$$

satisfied by the function  $y(t)$ . Let

$$\delta^{(r)}(t) = x^{(r)}(t) - y^{(r)}(t) \quad (5.6)$$

denote the error of the approximation. Since the coefficients  $\hat{b}_{j-s}$  are functions of tabular data, we can use the relations (5.1)-(5.6) to express  $\delta^{(r)}(t)$  as a function of  $x(t), t=0, \Delta, \dots$ , and, thus, obtain the variance of  $\delta^{(r)}(t)$  expressed in terms of statistics of  $\varepsilon_t$ . This yields a measure of the accuracy of numerical derivatives obtained by the method described in the preceding sections.

In the present section we derive an approximation to the bound of the variance of  $\delta^{(r)}(t)$ . To this end let

$$\beta_{j-s} = \hat{b}_{j-s}(r) - b_{j-s}(r), \quad (5.7)$$

i.e.  $\beta_{j-s}$  is the error in the estimate of  $b_{j-s}(r)$ . In order to simplify subsequent algebraic expressions we assume that the estimates  $\hat{b}_{j-s}(r)$  are unbiased, i.e.

$$E(\beta_{j-s}) = 0. \quad (5.8)$$

By substituting (5.2) and (5.3) in (5.6) we get

$$\delta^{(r)}(t) = A_t + B_t, \quad (5.9)$$

where

$$A_t = \sum_{j=s+1}^{k+s} b_{j-s}(r) \varepsilon_{t+jq\Delta} \quad (5.10)$$

and

$$B_t \sum_{j=s+1}^{k+s} \beta_{j-s} x(t+jq\Delta). \quad (5.11)$$

It follows from (5.8) - (5.11) that  $E(\delta^{(r)}(t))=0$ . Therefore by squaring both sides of (5.9) and taking expected values we get

$$\sigma^2(r) = E(A_t^2) + E(B_t^2) + 2E(A_t B_t). \quad (5.12)$$

where  $\sigma^2(r)$  is the variance of the error in the  $r$ -th derivative.

Since  $E(B_t A_t) \leq \sqrt{E(A_t^2) E(B_t^2)}$  we get from (5.12):

$$\sigma^2(r) \leq \left( \sqrt{E(A_t^2)} + \sqrt{E(B_t^2)} \right)^2. \quad (5.13)$$

Thus,  $E(A_t^2)$  and  $E(B_t^2)$  define a bound on  $\sigma^2(r)$ .

It follows from (5.10) that

$$E(A_t^2) = \sigma^2 \sum_{j=s+1}^{k+s} b_{j-s}^2(r). \quad (5.14)$$

We replace the values  $b_{j-s}(r)$  in (5.14) by their estimates and obtain an approximation:

$$E(A_t^2) = \sigma^2 \sum_{j=s+1}^{k+s} \hat{b}_{j-s}^2(r). \quad (5.15)$$

Let  $\lambda_j$  and  $\hat{\lambda}_j$ ,  $j=1, 2, \dots, k$  be the eigenvalues of the difference equations (5.5) and (5.4), respectively. Let

$$\alpha_j = \hat{a}_j - a_j \quad (5.16)$$

and

$$\eta_j = \hat{\lambda}_j - \lambda_j. \quad (5.17)$$

We assume that all the eigenvalues of (5.4) and (5.5) are of multiplicity one. Then the coefficients  $b_{j-s}(r)$  and  $\hat{b}_{j-s}(r)$  are given by (4.19) in terms of  $\lambda_j$ 's and  $a_j$ 's and  $\hat{\lambda}_j$ 's and  $\hat{a}_j$ 's, respectively. An

expression for  $B_t$  in terms of the  $\alpha_j$ 's can be obtained from (4.19), (5.7), (5.16), and (5.17). This is a non-linear algebraic function. We approximate this function by the following linear relation

$$B_t = \sum_{j=s+1}^{k+s} x(t+jq\Delta) \sum_{i=1}^k \frac{d\beta_{j-s}}{d\alpha_i} \alpha_i \quad \text{or}$$

$$B_t = \sum_{i=1}^k \alpha_i \sum_{j=s+1}^{k+s} \frac{d\beta_{j-s}}{d\alpha_i} x(t+jq\Delta), \quad (5.18)$$

where

$$\frac{d\beta_{j-s}}{d\alpha_i} = \frac{d\beta_{j-s}}{d\alpha_i} \Big|_{\alpha_i=0} = \sum_{p=1}^k \frac{\partial \beta_{j-s}}{\partial \hat{\lambda}_p} \Big|_{\hat{\lambda}_p = \lambda_p} \cdot \frac{\partial \hat{\lambda}_p}{\partial \alpha_i} \Big|_{\alpha_i=0}$$

$$+ \frac{\partial \beta_{j-s}}{\partial \alpha_i} \Big|_{\alpha_i=0}, \quad (5.19)$$

and

$$\frac{\partial \beta_{j-s}}{\partial \alpha_i} \Big|_{\alpha_i=0} = \frac{\partial \beta_{j-s}}{\partial \hat{a}_i} \Big|_{\hat{a}_i = a_i}.$$

In order to simplify further derivation of an expression for  $E(B_t^2)$  we introduce the following notation:

$$N_p = \lambda_p^{-s} \log^r \lambda_p, \quad (5.20)$$

$$N_{pj} = \sum_{u=0}^{k-j+s} a_{k-j+s-u} \lambda_p^u, \quad (5.21)$$

$$D_p = \sum_{u=1}^k u \lambda_p^u a_{k-u}, \quad (5.22)$$

$$N(j,p) = N_p N_{pj}, \quad (5.23)$$

and

$$f(\lambda_1, \lambda_2, \dots, \lambda_k, a_1, a_2, \dots, a_k) = \frac{N(j,p)}{D_p} \quad (5.24)$$

Then it follows from (4.19) and (5.7) that  $\beta_{j-s} = f(\hat{\lambda}_1, \dots, \hat{\lambda}_k, \hat{a}_1, \dots, \hat{a}_k) - f(\lambda_1, \dots, \lambda_k, a_1, \dots, a_k)$ . Therefore,

$$\left. \frac{\partial \beta_{j-s}}{\partial \lambda_p} \right|_{\hat{\lambda}_p = \lambda_p} = \frac{D_p \frac{\partial N(j, p)}{\partial \lambda_p} - N(j, p) \frac{\partial D_p}{\partial \lambda_p}}{D_p^2} . \quad (5.25)$$

By (5.23)

$$\frac{\partial N(j, p)}{\partial \lambda_p} = \frac{\partial N_p}{\partial \lambda_p} N_{pj} + \frac{\partial N_{pj}}{\partial \lambda_p} N_p . \quad (5.26)$$

By (5.20)

$$\frac{\partial N_p}{\partial \lambda_p} = - \frac{s N_p}{\lambda_p} + \frac{r N_p}{\lambda_p \log \lambda_p} . \quad (5.27)$$

By (5.21)

$$\frac{\partial N_{pj}}{\partial \lambda_p} = \sum_{u=0}^{k-j+s} u a_{k-j+s-u} \lambda_p^{u-1} . \quad (5.28)$$

By (5.22)

$$\frac{\partial D_p}{\partial \lambda_p} = \sum_{u=1}^k u^2 \lambda_p^{u-1} a_{k-u} . \quad (5.29)$$

Similarly we get

$$\left. \frac{\partial \beta_{j-s}}{\partial a_i} \right|_{\hat{a}_i = a_i} = \sum_{p=1}^k \frac{\frac{\partial N(j, p)}{\partial a_i} D_p - \frac{\partial D_p}{\partial a_i} N(j, p)}{D_p^2} , \quad (5.30)$$

where

$$\frac{\partial N(j, p)}{\partial a_i} = N_p \lambda_p^{k-j+s-i} \quad (5.31)$$

and

$$\frac{\partial D_p}{\partial a_i} = (k-i) \lambda_p^{k-i} . \quad (5.32)$$

By definition of  $\lambda_p$  we have  $\lambda_p^k = \sum_{u=1}^k a_u \lambda_p^{k-u}$ . We differentiate this

expression with respect to  $a_i$  and solve the resulting equation for  $\partial \lambda_p / \partial a_i$ :

$$\frac{\partial \lambda_p}{\partial a_i} = \lambda_p^{k-i} / \left[ k \lambda_p^{k-1} - \sum_{u=1}^k (k-u) a_u \lambda_p^{k-u-1} \right]. \quad (5.33)$$

The formulas (5.19) - (5.33) define the coefficients  $d\beta_{j-s} / d\alpha_i$  in (5.18) and, thus, the quantities  $C_i$  in the following:

$$C_i = \sum_{j=s+1}^{k+s} \frac{d\beta_{j-s}}{d\alpha_i} x(t+jq\Delta). \quad (5.34)$$

We have by (5.18)  $B_t = \sum_{i=1}^k C_i \alpha_i$ . Hence  $B_t^2 = \sum_{v=1}^k \sum_{u=1}^k \alpha_u \alpha_v C_u C_v$  and

$$E(B_t^2) = E_t \left[ E(B_t^2 | t) \right] = E_t \left[ \sum_{v=1}^k \sum_{u=1}^k C_u C_v E(\alpha_u \alpha_v | t) \right], \quad (5.35)$$

where the conditional expectation is over the distribution density function of the  $\alpha_i$ 's. Since  $E(\alpha_u \alpha_v | t)$  does not depend on  $t$  we get

$$E(B_t^2) = \sum_{v=1}^k \sum_{u=1}^k E(\alpha_u \alpha_v). \quad E_t(C_u C_v) = \sum_{v=1}^k \sum_{u=1}^k E(\alpha_u \alpha_v) \cdot \frac{1}{N} \sum_t C_u(t) C_v(t) = \frac{1}{N} \sum_t \sum_{v=1}^k \sum_{u=1}^k E(\alpha_u \alpha_v) C_u(t) C_v(t).$$

The quantity  $\tilde{N}\sigma_p^2$  in (2.17) is small as compared to the diagonal elements of the matrix  $M$ . Therefore the estimates  $\hat{a}_j$  in (5.4) differ only little from the least squares estimates of the  $a_j$ 's, and hence the covariance matrix  $(E(\alpha_u \alpha_v))$  can be approximated by  $\sigma^2 (M - \tilde{N}\sigma^2 I)^{-1} \equiv \sigma^2 \tilde{M}$ . By writing  $m_{uv}$  for an element of  $\tilde{M}$  obtained by estimating the regression coefficients ( $a_j$ 's) we have the following approximation:

$$E(B_t^2) = \frac{\sigma^2}{N} \sum_t \sum_{u=1}^k \sum_{v=1}^k C_u(t) C_v(t) m_{uv}. \quad (5.36)$$

If  $\lambda_p$ 's and  $a_j$ 's in the definition of  $C_u$  are replaced by  $\hat{\lambda}_p$ 's and  $\hat{a}_j$ 's, formula (5.36) yields an estimate of  $E(B_t^2)$ . This together with (5.13) and (5.15) defines an upper bound of the variance of the error in the estimate of the  $r$ -th derivative. However, this is not an absolute bound since its estimate depends on the assumption that (4.19) yields unbiased estimates of the  $b_j$ 's, on the accuracy of linearized expressions of  $\hat{b}_j$ 's in terms of  $\hat{a}_j$ 's and  $\hat{\lambda}_j$ 's, and on the effect of replacing  $a_j$ 's and  $\lambda_j$ 's by their estimates,  $\hat{a}_j$ 's and  $\hat{\lambda}_j$ 's, respectively.

A few examples listed in Appendix B show that the formula (5.13) yields reasonable bounds of  $\sigma^2(r)$  in almost all the cases tested and hence it can be used as an indicator of accuracy for the numerical estimates of derivatives obtained by the method of this report.

Further applications and examples will be presented in a companion report now in preparation. Additional sets of synthetic data as well as samples of field test data will be considered therein.

## REFERENCES

1. T.F. Dolgopolova and V.K. Ivanov, On Numerical Differentiation. *Zh. Vychisl Mat. Fiz.* 6, 3, 570-576, 1966.
2. R.S. Anderssen and P. Bloomfield, "A Time Series Approach to Numerical Differentiation." *Technometrics* 16, 1 69-75, 1974.
3. C. Masaitis, G. Francis and V. Woodward, "Survival vs. Horsepower per Ton Test Data Analysis," BRL MR 2518, 1975. (AD #C003211L)
4. R.F. Lieske and A.M. MacKenzie, "Determination of Aerodynamic Drag from Radar Data," BRL MR 2210, 1972.
5. F. Yagi, B. Jansen, L.B. Kennedy, W.C. Taylor, "Analysis of Interferometer Records of Projectile Motion in the Bore of Small Arms," BRL Report 939, 1955. (AD #74976)
6. D.B. Hunter, "An Iterative Method of Numerical Differentiation," *Comp. J.* 3, 270-271, 1960.
7. A. Talmi and G. Gilat, "Method for Smooth Approximation of Data," *Journal of Comp. Physics* 23, 93-123, 1977.
8. H.C. Hershey, J.L. Zakin and R. Simha, "Numerical Differentiation of Equally Spaced and Not Equally Spaced Experimental Data," *Ind. Eng. Chem. Fundam.* 6, 413-421, 1967.
9. P. Craven and G. Wahba, "Smoothing Noisy Data with Spline Functions," *Numer. Math.* 31, 377-403, 1979.
10. A.N. Tikhonov, "Solution of Incorrectly Formulated Problems and the Regularization Method," *Soviet Math. Dokl.* 4, 1035-38, 1963.
11. J. Cullum, "Numerical Differentiation and Regularization," *Siam J. Numer. Anal.* 8, 254-265, 1971.
12. R.S. Anderssen and P. Bloomfield, "Numerical Differentiation Procedures for Non-Exact Data," *Numer. Math.* 22, 157-182, 1974.
13. G.E. Box and G.M. Jenkins, *Time Series Analysis: Forecasting and Control.* Revised Edition. Holden-Day, 1976.

## APPENDIX A

Lemmas 1, 2, 3 below simplify proofs in Section 2, while Lemmas 4, 5 are found useful in Section 3.

Lemma 1. Suppose  $y(t)$  satisfies the conditions (a), (b), and (c) of Section 2. Let  $S_j(\Delta)$  be the symmetric function of the roots of (2.4) of order  $j$  and  $M_j(\Delta) = S_j(\Delta) - \binom{k}{j}$ . Let

$$N(\Delta) \equiv \sum_{j=1}^k (-1)^{j-1} M_j(\Delta) y(t-j\Delta). \quad (A.1)$$

Then

$$\left. \frac{d^q N(\Delta)}{d\Delta^q} \right|_{\Delta=0} = 0 \quad \text{for } q=0, 1, \dots, k-1. \quad (A.2)$$

Proof. With  $f(t)$  replaced by  $y(t)$  equation (2.4) can be written as follows by expanding  $P_\Delta$ , replacing powers of  $B_\Delta$ , and solving for  $y(t)$ :

$$y(t) = \sum_{j=1}^k (-1)^{j-1} S_j(\Delta) y(t-j\Delta). \quad (A.3)$$

By substituting (A.3) into the identity

$$(1-B_\Delta)^k y(t) = y_t + \sum_{j=1}^k (-1)^j \binom{k}{j} t(t-j\Delta) \text{ we obtain}$$

$$(1-B_\Delta)^k y(t) = \sum_{j=1}^k (-1)^{j-1} M_j(\Delta) y(t-j\Delta) \equiv N(\Delta). \quad (A.4)$$

It follows from (A.4) that the  $k$ -th derivative of  $y(t)$  is

$$y^{(k)}(t) = \lim_{\Delta \rightarrow 0} \frac{N(\Delta)}{\Delta^k}. \quad (A.5)$$

Now we prove (A.2) by induction on  $q$ . Indeed, it follows from the definition of  $M_j(\Delta)$  that  $M_j(0)=0$  since each term of  $S_j(0) = 1$ . Hence by (A.1)  $N(0)=0$ , i.e. (A.2) holds for  $q=0$ . Suppose it holds for  $q=n < k-1$ . Then by applying L'Hospital's rule to (A.5)  $n+1$  times we get

$$y^{(k)}(t) = \lim_{\Delta \rightarrow 0} \frac{d^{n+1}N(\Delta)/d\Delta^{n+1}}{k(k-1)\dots(k-n)\Delta^{k-n-1}} \quad (A.6)$$

By hypothesis (b)  $y^{(k)}(t)$  is bounded. Therefore (A.6) implies that  $d^{n+1}N(\Delta)/d\Delta^{n+1} \Big|_{\Delta=0} = 0$ , i.e. (A.2) holds for  $q = n+1 < k$ . Q.E.D.

For  $n_j$  and  $\lambda_j$  in (2.2) let  $m_j = \sum_{i=1}^{j-1} (n_i + 1)$  and  $v_{m_j+1} = v_{m_j+2} = \dots = v_{m_{j+1}} = \lambda_j$ . Let  $T_{k,j}$  be the set of all the combinations of the integers  $1, 2, \dots, k$  taken  $j$  at a time. Let  $M_j = M_j(\Delta)$ .

Lemma 2. Let

$$L_k(i_1, i_2, \dots, i_j) = \sum_{u=1}^j \log v_{i_u} \quad (A.7)$$

Then

$$\sum_{j=1}^k (-1)^j j^p \sum_{T_{k,j}} L_k^{q-p}(i_1, \dots, i_j) = 0 \quad (A.8)$$

for  $p < q < k$ .

Proof. Differentiation of (A.1) yields

$$\begin{aligned} \frac{d^q N(\Delta)}{d\Delta^q} &= \sum_{j=1}^k (-1)^{j-1} \left[ M_j y^{(q)}(t-j\Delta) (-j)^q \right. \\ &\quad \left. + \sum_{p=0}^{q-1} \binom{q}{p} M_j^{(q-p)} y^{(p)}(t-j\Delta) (-j)^p \right]. \end{aligned} \quad (A.9)$$

But

$$M_j^{(q-p)} = \sum_{T_{k,j}} v_{i_1} \Delta v_{i_2} \Delta \dots v_{i_j} \Delta L_k^{q-p}(i_1, i_2, \dots, i_j).$$

Thus,

$$\frac{d^q N(\Delta)}{d\Delta^q} \Big|_{\Delta=0} = \sum_{j=1}^k (-1)^{j-1} \sum_{p=0}^{q-1} y^{(p)}(t) (-j)^p \cdot \sum_{T_{k,j}} \binom{q}{p} L_k^{q-p}(i_1, i_2, \dots, i_j).$$

By interchanging the order of summation we get

$$\frac{d^q N(\Delta)}{d\Delta^q} \Big|_{\Delta=0} = \sum_{p=0}^{q-1} y^{(p)}(t) \binom{q}{p} \sum_{j=1}^k (-1)^{j-1-p} j^p \cdot \sum_{T_{k,j}} L_k^{q-p}(i_1, i_2, \dots, i_j). \quad (A.10)$$

It follows from (A.2) and (A.10) that

$$\sum_{p=0}^{q-1} y^{(p)}(t) \binom{q}{p} \sum_{j=1}^k (-1)^{j-1-p} j^p \sum_{T_{k,j}} L_k^{q-p}(i_1, i_2, \dots, i_j) = 0 \quad (A.11)$$

for  $q < k$ .

Let  $p_0 \leq q-1$  be the largest value of  $p$  with non-vanishing coefficient of  $y^{(p)}$  in (A.11), i.e. (A.11) can be written in the form

$$y^{(p_0)} + \sum_{p=0}^{p_0-1} \tilde{c}_p y^{(p)} = 0, \text{ an ordinary differential equation of order } p_0$$

whose solution  $y(t)$  is thus of the form  $y(t) = \sum_{j=1}^m \sum_{i=0}^{\tilde{n}_j} \tilde{c}_{ji} t^i e^{\mu_j t}$

with  $p_0 = \sum_{j=1}^{\tilde{m}} (\tilde{n}_j + 1)$ . But it then follows that  $y(t)$  satisfies the

difference equation  $P(B)y(t)=0$  where  $P(\lambda) = \sum_{j=1}^{\tilde{m}} (\lambda - e^{\mu_j})^{\tilde{n}_j + 1}$ , i.e. a

difference equation of order  $p_0 \leq q-1 < k$ , contrary to the assumption (c). Therefore in (A.11) we must have

$$\sum_{j=1}^k (-1)^j j^p \sum_{T_{k,j}} L_k^{q-p}(i_1, \dots, i_j) = 0, \quad (A.12)$$

for  $p < q < k$ , which is (A.8). Q.E.D.

Next we prove the following lemma, making use of the previous one.

Lemma 3. Let  $L_k(i_1, i_2, \dots, i_j)$  be as in (A.7). Then

$$\tilde{S}_{k,u} = \frac{(-1)^k}{u!(k-u)!} \sum_{j=1}^k (-1)^j j^{k-u} \sum_{T_{k,j}} L_k^u(i_1, i_2, \dots, i_j) \equiv V \quad (A.13)$$

for  $u=1, 2, \dots, k$ , where  $\tilde{S}_{k,u}$  is the symmetric function of  $\xi_i = \log v_i$ ,  $i=1, 2, \dots, k$ , of order  $u$ .

Proof. This lemma is proved by induction on  $k$ . When  $k=1$  the right hand side of (A.13) is  $-1 \cdot (-1) \cdot \xi_1$  which is  $\tilde{S}_{1,1}$ . Suppose that (A.13) holds for  $k-1$ . For any positive integer  $u$ ,  $\tilde{S}_{k,u}$  is a linear function in each argument and it can be written as follows:

$$\tilde{S}_{k,u}(\xi_k) = \xi_k \tilde{S}_{k-1,u-1} + \tilde{S}_{k-1,u} \quad (A.14)$$

for  $u=1, 2, \dots, k$  (we assume that  $\tilde{S}_{k-1,k} = 0$  and  $\tilde{S}_{k,0} = 1$  for  $k \geq 1$ ). Since  $V$  in (A.13) is a symmetric function of the  $\xi_i$ 's, we need only to show that  $V$  is representable in the form (A.14), i.e. that

$$V \Big|_{\xi_k=0} = \tilde{S}_{k-1,u} \quad (A.15)$$

and

$$\frac{\partial V}{\partial \xi_k} = \tilde{s}_{k-1, u-1} . \quad (A.16)$$

We write (A.13) as follows:

$$\begin{aligned} V(\xi_k) = & \frac{(-1)^k}{u!(k-u)!} \left\{ - \left( \sum_{T_{k-1,1}} L_{k-1}^u(i) + \xi_k^u \right) \right. \\ & + (-1)^k k^{k-u} L_{k-1}^u (1, 2, \dots, k-1) \\ & + \sum_{j=2}^{k-1} (-1)^j j^{k-u} \left[ \sum_{T_{k-1,j}} L_{k-1}^u (i_1, \dots, i_j) \right. \\ & \left. \left. + \sum_{T_{k-1,j-1}} (L_{k-1}(i_1, i_2, \dots, i_{j-1}) + \xi_k)^u \right] \right\} . \quad (A.17) \end{aligned}$$

Therefore,

$$\begin{aligned} V(0) = & \frac{(-1)^k}{u!(k-u)!} \left\{ - \sum_{T_{k-1,1}} L_{k-1}^u(i) + (-1)^k k^{k-u} L_{k-1}^u (1, 2, \dots, k-1) \right. \\ & + \sum_{j=2}^{k-1} (-1)^j j^{k-u} \sum_{T_{k-1,j}} L_{k-1}^u (i_1, i_2, \dots, i_j) \\ & \left. + \sum_{j=2}^{k-1} (-1)^j j^{k-u} \sum_{T_{k-1,j-1}} L_{k-1}(i_1, i_2, \dots, i_{j-1}) \right\} . \end{aligned}$$

By changing the summation index in the last sum and by combining the terms we get

$$\begin{aligned}
 v(0) = & \frac{(-1)^k}{u!(k-u)!} \left\{ (2^{k-u-1}) \sum_{T_{k-1,1}} L_{k-1}^u(i) \right. \\
 & + \sum_{j=2}^{k-2} (-1)^j \left[ j^{k-u} - (j+1)^{k-u} \right] \cdot \sum_{T_{k-1,j}} L_{k-1}^u(i_1, i_2, \dots, i_j) \\
 & \left. + (-1)^k \left[ k^{k-u} - (k-1)^{k-u} \right] \cdot L_{k-1}^u(1, 2, \dots, k-1) \right\}. \quad \text{We apply the}
 \end{aligned}$$

binomial theorem to the expressions in the brackets and get

$$\begin{aligned}
 v(0) = & \frac{(-1)^{k-1}}{u!(k-u)!} \left\{ \sum_{j=1}^{k-1} (-1)^j (k-u) j^{k-u-1} \sum_{T_{k-1,j}} L_{k-1}^u(i_1, \dots, i_j) \right. \\
 & \left. + \sum_{j=1}^{k-1} (-1)^j \sum_{i=2}^{k-u} \binom{k-u}{i} j^{k-u-i} \sum_{T_{k-1,j}} L_{k-1}^u(i_1, \dots, i_j) \right\}.
 \end{aligned}$$

By the inductive assumption the first term in the braces multiplied by the factor outside the braces is  $\tilde{S}_{k-1,u}$ . We interchange the order of summation in the second term and get:

$$\begin{aligned}
 v(0) = & \tilde{S}_{k-1,u} + \frac{(-1)^{k-1}}{u!(k-u)!} \sum_{i=2}^{k-u} \binom{k-u}{i} \\
 & \cdot \left[ \sum_{j=1}^{k-1} (-1)^j j^{k-u-i} \sum_{T_{k-1,j}} L_{k-1}^u(i_1, \dots, i_j) \right].
 \end{aligned}$$

By (A.8) the expression in the brackets is zero, i.e.  $v(0) = \tilde{S}_{k-1,u}$ . Thus, (A.15) is proved.

In order to verify (A.16) we differentiate (A.17) with respect to  $\xi_k$  and get:

$$\begin{aligned} \frac{\partial V}{\partial \xi_k} = & \frac{(-1)^k}{u!(k-u)!} \left\{ -u \xi_k^{u-1} + u(-1)^k k^{k-u} \left[ L_{k-1}(1, 2, \dots, k-1) + \xi_k \right]^{u-1} \right. \\ & \left. + \sum_{j=2}^{k-1} (-1)^j j^{k-u} \sum_{T_{k-1, j-1}} u \left[ L_{k-1}(i_1, \dots, i_{j-1}) + \xi_k \right]^{u-1} \right\}. \end{aligned} \quad (A.18)$$

By applying the binomial theorem to the expressions in the brackets and by combining the terms containing the same powers of  $\xi_k$  we get:

$$\begin{aligned} \frac{\partial V}{\partial \xi_k} = & \frac{(-1)^k}{(u-1)!(k-u)!} \left\{ \sum_{j=1}^{k-1} (-1)^{j+1} (j+1)^{k-u} \sum_{T_{k-1, j}} L_{k-1}^{u-1} (i_1, \dots, i_j) \right. \\ & + \left[ -1 + \sum_{j=2}^k (-1)^j j^{k-u} \binom{k-1}{j-1} \right] \xi_k^{u-1} \\ & \left. + \sum_{i=1}^{u-2} \binom{u-1}{i} \xi_k^i \sum_{j=2}^k (-1)^j j^{k-u} \sum_{T_{k-1, j-1}} L_{k-1}^{u-1-i} (i_1, \dots, i_{j-1}) \right\}. \end{aligned} \quad (A.19)$$

By applying the binomial theorem again and by interchanging the order of summation the coefficient of the zero power of  $\xi_k$  in the braces can be written as follows:

$$- \sum_{i=0}^{k-u} \binom{k-u}{i} \sum_{j=1}^{k-1} (-1)^j j^{k-u-i} \sum_{T_{k-1, j}} L_{k-1}^{u-1} (i_1, \dots, i_j).$$

By (A.8) all the terms of the outer summation except the term corresponding to  $i=0$  vanish. Hence the coefficient of the zero power of  $\xi_k$  in (A.18) is

$$\frac{(-1)^{k-1}}{(u-1)!(k-u)!} \sum_{j=1}^{k-1} (-1)^j j^{k-u} \sum_{T_{k-1, j}} L_{k-1}^{u-1} (i_1, \dots, i_j).$$

By the inductive assumption this is  $\tilde{S}_{k-1, u-1}$ . Thus, in order to complete the proof of (A.16) it remains to show that the coefficients of every non-zero power of  $\xi_k$  in (A.19) vanish. Since the highest power of  $\xi_k$  in (A.19) is  $u-1$  we need to consider only the case of  $u > 1$ .

The coefficient of  $\xi_k^{u-1}$  can be written as a constant times the form:

$$\begin{aligned} \sum_{j=1}^k (-1)^j j^{k-u} \binom{k-1}{j-1} &= - \sum_{i=0}^{k-1} (-1)^i (i+1)^{k-u} \binom{k-1}{i} \\ &= -(1-B)^{k-1} x^{k-u} \Big|_{x=k} . \end{aligned}$$

Since  $u > 1$  it follows that the coefficient of  $\xi_k^{u-1}$  is zero. The coefficients of  $\xi_k^i$ ,  $i=1, 2, \dots, u-2$  in (A.19) vanish because of (A.8). This completes the proof of (A.13).

We use the following notation in the next lemma:  $\prod_{v=0}^{-1} (r-v) = 1$ ,

$$\Lambda_{qp}^T = \left( (s+1)^p \lambda_q^{s+1}, (s+2)^p \lambda_q^{s+2}, \dots, (s+k)^p \lambda_q^{s+k} \right),$$

$$\Lambda^T = \frac{1}{\Delta^r} \left( \log^{r-p} \lambda_q \prod_{v=0}^{p-1} (r-v) \right) \text{ for } p=0, 1, 2, \dots, n_q, q=1, 2, \dots, m,$$

$$\text{and } L = (\Lambda_{10}, \Lambda_{11}, \dots, \Lambda_{1n_1}, \Lambda_{20}, \Lambda_{21}, \dots, \Lambda_{2n_2}, \dots, \Lambda_{m,n_m})^T.$$

Lemma 4. Let  $y(t) \in A$ , i.e.

$$y(t) = \sum_{j=1}^m \sum_{i=0}^{n_j} c_{ji} t^i \lambda_j^t \quad (A.20)$$

with  $k = \sum_{j=1}^m (n_j + 1)$ . If there exist  $b^T = (b_1, b_2, \dots, b_k)$  such that

$$Lb = \Lambda \quad (A.21)$$

then for every integer  $s$  satisfying  $-t - \Delta \leq s\Delta \leq T - k\Delta$

$$y^{(r)}(t) = \sum_{j=s+1}^{s+k} b_{j-s} y(t+j\Delta). \quad (A.22)$$

Proof. Substitute (A.20) on both sides of (A.22) and equate the coefficients of the terms  $t^i \lambda_j^t$  on the opposite sides of the result. Then (A.21) is obtained. The lemma is proved by reversing these steps as follows.

The expression (A.21) is a system of equations

$$\Lambda_{qp}^T b = \frac{1}{\Delta^r} \log^{r-p} \lambda_q \prod_{v=0}^{p-1} (r-v), \quad p=0, 1, \dots, n_q, \quad q=1, 2, \dots, m \text{ or}$$

$$\sum_{j=1}^k (s+j)^p \lambda_q^{s+j} b_j = \frac{1}{\Delta^r} \log^{r-p} \lambda_q \prod_{v=0}^{p-1} (r-v). \quad (A.23)$$

For each  $q=1, 2, \dots, m$  and  $z=1, 2, \dots, n_q$  multiply the equations of (A.23) corresponding to  $p=u-z$  ( $u=z, z+1, \dots, n_q$ ) by  $c_{qu} \binom{u}{z}$  and sum on  $u$ . This yields:

$$\begin{aligned} & \sum_{u=z}^{n_q} c_{qu} \binom{u}{z} \sum_{j=1}^k (s+j)^{u-z} \lambda_q^{s+j} b_j \\ &= \frac{1}{\Delta^r} \sum_{u=z}^{n_q} c_{qu} \binom{u}{z} \log^{r-u+z} \lambda_q \prod_{v=0}^{u-z-1} (r-v). \quad (A.24) \end{aligned}$$

Since  $\binom{u}{z} \prod_{v=0}^{u-z-1} (r-v) = \binom{u}{u-z} \prod_{v=0}^{u-z-1} (r-v) = \binom{r}{u-z} \prod_{v=0}^{u-z-1} (u-v)$  we can

write (A.24) in the form

$$\begin{aligned} & \sum_{u=z}^{n_q} c_{qu} \binom{u}{u-z} \sum_{j=1}^k (s+j)^{u-z} \lambda_q^{s+j} b_j \\ &= \frac{1}{\Delta^r} \sum_{u=z}^{n_q} c_{qu} \log^{r-u-z} \lambda_q \binom{r}{u-z} \prod_{v=0}^{u-z-1} (u-v). \quad (A.25) \end{aligned}$$

Now we multiply (A.25) by  $t^z$  and sum on  $z$  from 0 to  $n_q$ :

$$\begin{aligned} & \sum_{z=0}^{n_q} t^z \sum_{u=n_q}^z c_{qu} \binom{u}{u-z} \sum_{j=1}^k (s+j)^{u-z} \lambda_q^{s+j} b_j \\ &= \frac{1}{\Delta^r} \sum_{z=0}^{n_q} \sum_{u=z}^{n_q} c_{qu} t^z \log^{r-u-z} \lambda_q \binom{r}{u-z} \prod_{v=0}^{u-z-1} (u-v). \end{aligned}$$

We interchange the order of summation on both sides of this equality:

$$\begin{aligned} & \sum_{j=1}^k b_j \lambda_q^{s+j} \sum_{u=0}^{n_q} c_{qu} \sum_{z=0}^u \binom{u}{z} t^z (s+j)^{u-z} \\ &= \frac{1}{\Delta^r} \sum_{u=0}^{n_q} c_{qu} \sum_{z=0}^u t^z \log^{r-u-z} \lambda_q \binom{r}{u-z} \prod_{v=0}^{u-z-1} (u-v). \end{aligned}$$

Multiplication of both sides of this equality by  $\lambda_q^t$  yields:

$$\sum_{j=1}^k b_j \sum_{u=0}^{n_q} c_{qu} \lambda_q^{t+s+j} (t+s+j)^u = \sum_{u=0}^{n_q} c_{qu} \frac{d^r}{dt^r} (t^u \lambda_q^t).$$

By summing both sides of this equality on  $q$  from 1 to  $m$  we get:

$$\begin{aligned} & \sum_{j=1}^m b_j \sum_{q=1}^m \sum_{u=0}^{n_q} c_{qu} \lambda_q^{t+s+j} (t+s+j)^u \\ &= \sum_{q=1}^m \sum_{u=0}^{n_q} \frac{d^r}{dt^r} (c_{qu} t^u \lambda_q^t). \end{aligned}$$

In view of (A.20) this is  $\sum_{j=1}^k b_j y(t+s+j) = y^{(r)}(t)$  which is equivalent to (A.22).

Lemma 5. Denote  $n!! = \prod_{j=1}^n j!$ ,  $n := \sum_{j=1}^n j$ , and  $p_n = \prod_{j=1}^n \lambda_j^{(s+1)(n_j+1)}$ .

Let the matrix  $L$  be as in Lemma 4. Then

$$\det L = \prod_{i=1}^m n_i!! \prod_{j=1}^m \lambda_j^{n_j} : \prod_{j=2}^m \prod_{i=1}^{j-1} (\lambda_j - \lambda_i)^{(n_i+1)(n_j+1)}. \quad (A.26)$$

Proof. Let  $T_i = \mu_i \frac{\partial}{\partial \mu_i}$  and  $T_i^p (\lambda_j) f(\mu_i) = T_i^p f(\mu_i) \Big|_{\mu_i = \lambda_j}$ ,

i.e. the result of the operator  $T_i^p (\lambda_j)$  is obtained by applying the operator  $T_i$   $p$  times and then substituting  $\lambda_j$  for  $\mu_i$ . In particular,

$$T_i^{(0)} (\lambda_j) f(\mu_i) = f(\lambda_j). \quad \text{Let } m_j = \sum_{i=1}^j (n_i+1) \text{ and}$$

$$T(p) = \prod_{j=1}^p \prod_{i=m_{j-1}+1}^{m_j} T_i^{i-m_{j-1}-1} (\lambda_j). \quad (A.27)$$

Let

$$L^* = \begin{vmatrix} \mu_1^{s+1} & \mu_1^{s+2} & \dots & \mu_1^{s+k} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_k^{s+1} & \mu_k^{s+2} & \dots & \mu_k^{s+k} \end{vmatrix}.$$

Then  $T(m)L^* = \det L$ . This identity is used to prove the lemma by induction on  $n_j$ ,  $j=1,2,\dots,m$ . Indeed, if  $n_j=0$ ,  $j=1,2,\dots,m=k$  then by (A.27) the operator  $T(m) = T(k)$  simply replaces  $\mu_j$  by  $\lambda_j$ ,  $j=1,2,\dots,k$ , i.e. the left hand side of (A.26) is a Vandermonian multiplied by  $\prod_{j=1}^m \lambda_j^{s+1}$  and the right hand side reduces to  $\prod_{j=1}^m \lambda_j^{s+1} \prod_{j=2}^k \prod_{i=1}^{j-1} (\lambda_j - \lambda_i)$ ,

i.e. (A.26) holds for this choice of  $n_j$ 's. By induction assume that (A.26) holds for  $n_1, n_2, \dots, n_{p-1}$  with

$$n_{p+1} = n_{p+2} = \dots = n_m = 0 \quad (A.28)$$

and  $n_p$  replaced by  $v$  such that  $0 \leq v \leq n_p$ . Under these assumptions (A.27) implies that

$$T(m) = T^*(j, p, v) T(p, v), \quad (A.29)$$

where  $T^*(j, p, v)$  is an operator that replaces  $\mu_j$  by  $\lambda_{j-m_{p-1}-v+p-1}^{i-m_{p-1}-1}$  for

$j=m_{p-1}+v+1, \dots, k$  and

$$T(p, v) = \prod_{i=m_{p-1}+1}^{m_{p-1}+v+1} T_i^{i-m_{p-1}-1} T(p-1), \quad (A.30)$$

with  $T(p-1)$  defined by (A.27).

Let

$$P^*(p) = \prod_{i=1}^{p-1} n_i ! \prod_{i=1}^{p-1} \lambda_i^{n_i} \lambda_{p-1}^{p-1} \prod_{p-1 \geq j > i} (\lambda_j - \lambda_i)^{(n_i+1)(n_j+1)} \quad (A.31)$$

By induction (A.26) holds for the parameters in (A.28). Hence it follows:

$$\begin{aligned} T^*(j, p, q) T(p, q) L^* &= P^*(p) q ! \lambda_p^q : \lambda_p^{(q+1)(s+1)} \\ &\cdot \prod_{j=p+1}^m \lambda_j^{s+1} \prod_{i < p} (\lambda_p - \lambda_i)^{(q+1)(n_i+1)} \prod_{j > p} (\lambda_j - \lambda_p)^{q+1} \\ &\cdot \prod_{\substack{j > p \\ i \leq p-1}} (\lambda_j - \lambda_i)^{n_i+1} \prod_{j > i > p} (\lambda_j - \lambda_i) \end{aligned}$$

or

$$\begin{aligned} T^*(j, p, q) T(p, q) L^* &= P^*(p) q ! \lambda_p^q : \lambda_p^{(q+1)(s+1)} \\ &\cdot \prod_{i < p} (\lambda_p - \lambda_i)^{(q+1)(n_i+1)} T^*(j, p, q) \prod_{\substack{j > m_{p-1}+q+1 \\ i \leq p-1}} (\mu_j - \lambda_i)^{n_i+1} \\ &\cdot \prod_{j > i > m_{p-1}+q+1} (\mu_j - \mu_i) \prod_{j > m_{p-1}+q+1} \mu_j^{s+1} \prod_{j > p} (\mu_j - \lambda_p)^{q+1}. \end{aligned} \quad (A.32)$$

Since  $T^*(j, p, q)$  has an inverse it follows from (A.32) that

$$\begin{aligned}
 T(p, q)L^* &= P^*(p)q! \lambda_p^{q:} \prod_{i < p} (\lambda_p - \lambda_i)^{(q+1)(n_i+1)} \\
 &\cdot \prod_{j > i > m_{p-1}+q+1} (\mu_j - \mu_i) \prod_{j > m_{p-1}+q+1} \mu_j^{s+1} \prod_{\substack{j > m_{p-1}+q+1 \\ i \leq p-1}} (\mu_j - \lambda_i)^{n_i+1} \\
 &\cdot \prod_{j > m_{p-1}+q+1} (\mu_j - \lambda_p)^{q+1} \quad . \tag{A.33}
 \end{aligned}$$

Now let us apply the operator  $T_{m_{p-1}+q+2}^{q+1}(\lambda_p)$  to both sides of (A.33). It follows from (A.30) that

$$T_{m_{p-1}+q+2}^{q+1}(\lambda_p)T(p, q) = T(p, q+1). \tag{A.34}$$

We write the right hand side, say,  $R(p, q)$  of (A.33) in the following form:

$$R(p, q) = (\mu_{m_{p-1}+q+2} - \lambda_p)^{q+1} R_1, \tag{A.35}$$

where

$$\begin{aligned}
 R_1 &= P^*(p)q! \lambda_p^{q:} \prod_{i < p} (\lambda_p - \lambda_i)^{(q+1)(n_i+1)} \\
 &\cdot \prod_{j > m_{p-1}+q+2} (\mu_j - \mu_{m_{p-1}+q+2}) \prod_{j > i > m_{p-1}+q+2} (\mu_j - \mu_i) \mu_{m_{p+1}+q+2}^{s+1} \\
 &\cdot \prod_{j > m_{p-1}+q+2} \mu_j^{s+1} \prod_{i < p-1} (\mu_{m_{p-1}+q+2} - \lambda_i)^{n_i+1} \prod_{\substack{j > m_{p-1}+q+2 \\ i \leq p-1}} (\mu_j - \lambda_i)^{n_i+1} \\
 &\cdot \prod_{j > m_{p+1}+q+2} (\mu_j - \lambda_p)^{q+1}. \tag{A.36}
 \end{aligned}$$

Application of  $T_{m_{p-1}+q+2}^{q+1}$  to (A.35) yields a sum of terms containing

derivatives of  $(\mu_{m_{p-1}+q+2} - \lambda_p)^{q+1}$  with respect to  $\mu_{m_{p-1}+q+2}$  of orders

0 up to  $q+1$ . Thus, we can write

$$T_{m_{p-1}+q+2}^{q+1} R(p, q) = (q+1)! \mu_{m_{p-1}+q+2}^{q+1} R_1$$

$$+ (\mu_{m_{p-1}+q+2} - \lambda_p)^{q+1} R_2, \quad (A.37)$$

where the second term on the right hand side is the sum of the terms with derivatives of  $(\mu_{m_{p-1}+q+2} - \lambda_p)^{q+1}$  of order less than  $q+1$ . Therefore by substituting  $\mu_{m_{p-1}+q+2} = \lambda_p$  in (A.36) and in (A.37) we get

$$T_{m_{p-1}+q+2}^{q+1} (\lambda_p) R(p, q) = (q+1)! \lambda_p^{q+1} p^* (p) q! \lambda_p^q : \lambda_p^{(q+1)(s+1)}$$

$$\cdot \prod_{i \leq p-1} (\lambda_p - \lambda_i)^{(q+1)(n_i+1)} \prod_{j > m_{p-1}+q+2} (\mu_j - \lambda_p) \prod_{j > i > m_{p-1}+q+2} (\mu_j - \mu_i) \lambda_p^{s+1}$$

$$\cdot \prod_{j > m_{p-1}+q+2} \mu_j^{s+1} \prod_{i \leq p-1} (\lambda_p - \lambda_i)^{n_i+1} \prod_{j > m_{p-1}+q+2} (\mu_j - \lambda_i)^{n_i+1}$$

$$\cdot \prod_{j \geq m_{p-1}+q+2} (\mu_j - \lambda_p)^{q+1}.$$

or

$$T_{m_{p-1}+q+2}^{q+1} (\lambda_p) R(p, q) = (q+1)! \lambda_p^{(q+1)} : \prod_{i \leq p-1} (\lambda_p - \lambda_i)^{(q+2)(n_i+1)}$$

$$\cdot \prod_{j > m_{p-1}+q+2} (\mu_j - \lambda_p)^{q+2} p^* (p) \lambda_p^{(q+2)(s+1)} \prod_{j > m_{p-1}+q+2} \prod_{i \leq p-1} (\mu_j - \lambda_i)^{n_i+1}$$

$$\cdot \prod_{j \geq m_{p-1}+q+2} (\mu_j - \lambda_p)^{q+1} \prod_{j > i > m_{p-1}+q+2} (\mu_j - \mu_i) \prod_{j > m_{p-1}+q+2} \mu_j^{s+1}.$$

By comparing this with (A.33) we see that

$$T_{m_{p-1}+q+2}^{q+1} (\lambda_p) R(p, q) = R(p, q+1). \quad (A.38)$$

In view of (A.34) we have

$$T(p, q+1) L^* = R(p, q+1). \quad (A.39)$$

By applying the operator  $T^*(j, p, q+1)$  to both sides of (A.39) we obtain (A.26) for  $n_1, n_2, \dots, n_{p-1}$ , and  $n_p$  replaced by  $q+1$ . This completes the proof of the lemma.

## APPENDIX B

Details of the procedure for numerical differentiation according to the method of this report will be described in a forthcoming report that will contain examples of synthetic data as well as applications to experimental records. Here we include only a few illustrations that compare the accuracy of this method with the accuracy of moving polynomial arcs, Butterworth filter, and spline. These methods were selected for comparison because they are frequently applied to problems of interest to BRL. Furthermore, the corresponding computer programs are available. Also, spline approximation yields one of the most accurate, if not the most accurate, derivatives of all the commonly used methods for numerical differentiation. Comparison with Kalman filter is not included here since the results of this method are highly dependent on the assumed dynamics. If linear equations are assumed that are satisfied by a sine function of a given frequency, Kalman filter produces good results for the corresponding synthetic data. However, if the same dynamic model is applied to synthetic data obtained from, say, an exponential function, estimates of derivatives are poor. Illustration of these cases will be provided in the forthcoming report.

The synthetic data used here is described in Table 1. Here column 1 numbers the cases from 1 to 6 for convenience of reference. The corresponding functions  $x(t)$  are specified in column 2. The last case here is the Bessel function of the first kind of order zero. The values of the functions were computed at points in the intervals listed in column 4, and column 3 shows the step size for selecting the points in the respective intervals. Pseudorandom white gaussian noise with zero mean and the standard deviation  $\sigma$  given in column 5 was added to each value, and then various methods for numerical differentiation were applied to the noisy data.

Table 1. Synthetic Data

1 Case	2 $x(t)$	3 $\Delta$	4 $I$	5 $\sigma$
1	$\sin 2\pi t$	.004	[0,1]	.02
2	$\sin 2\pi t$	.004	[0,1]	.05
3	$e^t$	.01	[0,5]	.01
4	$4^t(1+t+t^2+t^3)$	.004	[0,2]	.32
5	$\sin 2\pi t + .1 \sin 10\pi t$	.004	[0,1]	.05
6	$J_0(t)$	.01	[1,6]	.01

The methods used are listed at the top of Table 2. Here moving polynomial arc corresponds to a cubic polynomial fitted to either 11 or 17 data points as indicated. The derivatives were evaluated at the midpoint of this span. Thus, derivatives at a few points at the beginning and the end of the data sequence are not available.

The Butterworth filter applied here corresponds to the transfer function  $B1 \pi/(s^3 + 2\pi s^2 + 2\pi^2 s + \pi^3)$ . This program does not provide first derivatives at 65 data points at the end of the data sequence when  $\Delta = .01$  and at 163 data points when  $\Delta = .004$ . Additional points are lost when higher derivatives are calculated. The method of this report provides derivatives at every data point with appropriate values of  $s$  in (3.2) and in the corresponding expressions for higher derivatives. The bulk of the derivatives are computed at the midpoint of the span

of formula (3.2), i.e., for  $s = -\left[\frac{k+1}{2}\right]$ . Estimates of the bounds as described in Section 5 also depend on  $s$ . In the examples of this appendix we use only  $s = -\left[\frac{k+1}{2}\right]$ , i.e., the RMSE's of the numerical derivatives are obtained here by comparing the derivatives computable by (3.2) with this value of  $s$  against the exact derivatives. Also an estimate of the bound for the standard deviation of the error is obtained by (5.13) for only these points.

The double rows labeled 1-6 in Table 2 show the percentage errors corresponding to the data sets 1-6 of Table 1 obtained by the methods shown at the top of Table 2. The RMS of the differences between exact and approximate first derivatives is divided by the RMS of the exact derivative and the ratio is multiplied by 100. This yields the percentage error of the first derivative for the respective method. This error is shown in the first row of a pair labeled  $x'$ . The second row of each pair labeled  $x''$  contains the percentage errors of the second derivative obtained in the same way. This table shows that the method of the report (current method) when applied to these synthetic data yields far more accurate derivatives than any of the other methods. For instance, in cases 1-3 the first derivative obtained by our method is from two to more than four times as accurate as by the second best method (spline), while the error in the second derivatives is lower by more than an order of magnitude. In the remaining cases the current method also produces much better results than other methods except for case 6 where derivatives obtained by the Butterworth filter are more accurate.

---

<sup>B1</sup> J.N. Groff and A.N. Gordon, "The Methodology and Preparatory Analysis of Tracking Data for the Antitank Missile Test (ATMT) Program. Part I: General Methodology and Shillelagh Analysis". Tech Report No. 151, AMSAA, 1976.

Table 2. Percentage Error

Method Case	Moving Polyn. Arc, 11 pts.	Moving Polyn. Arc, 17 pts.	Butterworth Filter	Spline	Current Method
1	X'	25	14	88	6
	X''	275	99	98	48
2	X'	62	33	88	15
	X''	688	248	98	120
3	X'	.46	.25	3.4	.010
	X''	13	4.7	6.9	2.2
4	X'	9	5	16	2.1
	X''	259	93	30	45
5	X'	55	30	91	13
	X''	89	255	99	46
6	X'	57	31	5.8	13
	X''	2341	8384	8.4	403
					33

Table 3 illustrates how an approximation of the bounds of the error compares with the actual RMSE's. Here the first row lists actual percentage errors of the first and second derivatives for the various cases listed in the headings of the column-pairs. This row is the same as the last column in Table 2. The second row here contains the ratios of the bounds  $\sigma(r)$  ( $r=1,2$ ) computed by (5.13) to the RMS of the respective exact values converted to percent. We see that (5.13) yields bounds about 2 to 10 times higher than the respective RMSE's except for the second derivative of case 6 (Bessel function) where the estimate of the bound is much too low.

Table 3. Error Bounds

1	2	3	4	5	6						
$X'$	$X''$										
1.3	1.4	3.4	3.5	.004	.004	.23	.80	8	18	7.8	33
3.5	2.4	8.9	6.2	.02	.02	2.1	3.0	29	91	10	4.5

## REFERENCE

B1. J.N. Groff and A.N. Gordon, "The Methodology and Preparatory Analysis of Tracking Data for the Antitank Missile Test (ATMT) Program. Part I: General Methodology and Shillelagh Analysis." Tech Report No. 151, AMSAA, 1976.

DISTRIBUTION LIST

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
12	Commander Defense Technical Info Center ATTN: DDC-DDA Cameron Station Alexandria, VA 22314	1	Director US Army Air Mobility Research and Development Laboratories Ames Research Center Moffett Field, CA 94035
1	Director Defense Advanced Research Project Agency Tactical Technical Office ATTN: Dr. James Tegnelia 1400 Wilson Boulevard Arlington, VA 22209	1	Commander US Army Communications Rsch and Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333	1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-TSS (2 cys) Dover, NJ 07801	1	Director US Army Electronics Command Atmospheric Sciences Laboratory ATTN: M.G. Heaps White Sands Missile Range, NM 88002
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L, Tech Lib Rock Island, IL 61299	2	Commander US Army Missile Command ATTN: DRSMI-R DRSMI-YDL Redstone Arsenal, AL 35809
1	Director US Army ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL Watervliet, NY 12189	1	Commander US Army Tank Automotive Rsch and Development Command ATTN: DRDTA-UL Warren, MI 48090
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-E 4300 Goodfellow Blvd. St. Louis, MO 63120	2	Commander US Army Natick Research and Development Command ATTN: DRXRE/Dr. D. Sieling Dr. E.W. Ross Natick, MA 01762

## DISTRIBUTION LIST

No. of Copies	<u>Organization</u>	No. of Copies	<u>Organization</u>
1	Commander US Army Research Office ATTN: CRD-AA-EH P.O. Box 12211 Research Triangle Park, NC 27709	3	AFGL (LKB, J. Paulson, K. Champion, W. Swider) Hanscom AFB, Ma 01731
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL, Tech Lib White Sands Missile Range NM 88002	1	AFRPL/LKCB (Dr. Horning) Edwards AFB, CA 93523
1	Director US Army BMD Advanced Technology Center P.O. Box 1500, West Station Huntsville, AL 35807	1	AFATL (Tech Lib) Eglin AFB, FL 32542
1	Commander US Army Concepts Analysis Agency 8120 Woodmont Avenue Bethesda, MD 20014	1	AFWL (SUL) Kirtland AFB, NM 87117
1	Deputy Under Secretary of the Army for Operations Research ATTN: Mr. David Hardison Washington, DC 20310	2	Director National Bureau of Standards US Department of Commerce ATTN: R.F. Hampson R. Kraft, Div 205.01 Washington, DC 20234
5	Commander Naval Research Laboratory ATTN: W. Ali D. Strobel J. Brown, Code 7700 E. Oran, Code 6020 Tech Lib, Code 2020 Washington, DC 20375	1	Director National Aeronautics and Space Administration Goddard Space Flight Center ATTN: R. Goldberg (Code 912) Greenbelt, MD 20771
1	USAF/AFRDDA Washington, DC 20311	1	Director NASA Scientific and Technical Information Facility ATTN: SAK/DL P.O. Box 8757 Baltimore/Washington International Airport, MD 21240
		1	Battelle Memorial Institute ATTN: Tech Lib 505 King Avenue Columbus, OH 43201

DISTRIBUTION LIST

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	University of Delaware Department of Mathematics ATTN: Prof. George Hdiao Newark, DE 19711		<u>Aberdeen Proving Ground</u>
1	University of Delaware College of Arts & Science ATTN: Prof. M.Z. Nashed 501 Kirkbride Office Bldg. Newark, DE 19711		Dir, USAMSA ATTN: DRXSY-D DRXSY-MP, H. Cohen Dr. A.W. Benton Dr. L.H. Crow Dr. E.M. Atzinger
1	University of Maryland ATTN: Prof. G.W. Stewart College Park, MD 20742		Cdr, USATECOM ATTN: DRSTE-T0-F
1	University of Maryland, Baltimore Campus Department of Mathematics ATTN: Dr. Thomas Seidman Baltimore, MD 21228		Dir, USACSL, Bldg E3516, EA ATTN: DRDAR-CLB-PA
2	University of Wisconsin - Madison Mathematics Research Center ATTN: J. Nohel, Director B. Noble 610 Walnut Street Madison, WI 53706		

## USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet, fold as indicated, staple or tape closed, and place in the mail. Your comments will provide us with information for improving future reports.

1. BRL Report Number \_\_\_\_\_

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.)  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.)  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: \_\_\_\_\_

Telephone Number: \_\_\_\_\_

Organization Address: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_